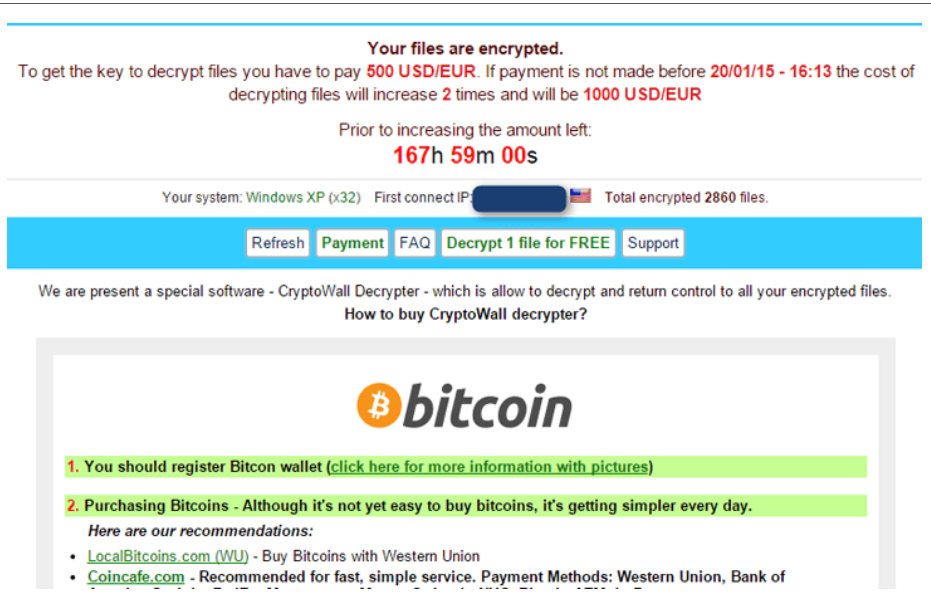
Defense against the Dark Arts
Overview / Terminology



malware

"evil software"

display a funny message

send passwords/credit card numbers to criminals

take pictures to send to criminals

delete data

hold data hostage

insert/replace ads in webpages

...

viruses

malware that inserts itself into another program

"infects" other programs when run

worms

independent program

usually "blends in" with system programs

copies itself to other machines or USB keys, etc.

configures systems to run it automatically

sometimes considered a kind of virus

trojan (horse)s

useful-looking program that is malware

e.g. looks like 'cracked' version of expensive commercial software

maybe is (or not), but also does something evil

potentially unwanted programs

unwanted (often malware-like) software bundled with wanted software

sometimes disclosed but in deceptive fine print

sometimes considered malware, sometimes not

rootkit

root = full privileges on a Unix-like system

rootkit = malware for obtaining full control of a system

rootkits usually evade removal, detection

e.g. program made invisible to "task manager"/`ps`

logic bomb

dormant malicious code

e.g. from disgruntled employee before quitting

vulnerabilities

trojans: the vulnerability is the user

otherwise?

software vulnerability

unintended program behavior that can be used by an adversary

vulnerability example

website able to install software without prompting

not intended behavior of web browser

software vulnerability classes (1)

memory safety bugs

- big topic in this course

"injection" bugs — type confusion

- commands/SQL within program

integer overflow/underflow

...

software vulnerability classes (2)

lack of checking inputs/permissions

```
http://webserver.com/../../../../
file-I-shouldn't-get.txt
```

almsot anything that's "undefined behavior" in C/C++

time-to-check to time-of-use

... more?

vulnerability versus exploit

exploit — something that uses a vulnerability to do something

proof-of-concept — something = demonstration the exploit is there

- example: open a calculator program

malware logistics: how?

what are they written in?

malware languages (1)

assembly language/machine code

- hand-coded or partially hand-coded

some vulnerabilities deal with machine code/memory layout

better for hiding malware from anti-malware tools

malware languages (2)

high-level scripting languages
- fast prototyping of vulnerabilities
- maintainability/efficiency usually not a priority
- vulnerabilities sometimes allow execution of malicious scripts

sometimes "toolkits" ('virus construction kit')
- by/for criminals to construct malware

Malware spreading

vulnerable network-accessible services

shared files/folders
- autorun on USB sticks
- macros in Word/Excel/etc. files

email attachments

websites + browser vulnerabilities
- JavaScript interpreter bugs
- Adobe Flash Player bugs

malware defenses (1)

"antivirus" software:

Windows Defender

avast!

Avira

AVG

McAfee

malware defenses (2)

app stores/etc. filtering (in theory)

"sandboxing" policies
- don't let, e.g., game access your taxes

some email spam filters

blacklists for web browsers
- Google Safe Browsing list (Chrome, Firefox)
- Microsoft SmartScreen (IE, Edge)

malware counter-defenses

malware authors tries to make it hard-to-detect

"obfuscation"

- make code (machine code/assembly/scripts/etc.) harder to read
- make code different each time (harder to blacklist)
- blend in with normal files/applications/etc.

New York Times

NEW YORK, FRIDAY, NOVEMBER 4, 1988 50 cents beyond 75 miles from New York City

'Virus' in Military Computers Disrupts Systems Nationwide

By JOHN MARKOFF

In an intrusion that raises questions about the vulnerability of the nation's computers, a Department of Defense network has been disrupted since Wednesday by a rapidly spreading "virus" program apparently introduced by a computer science student.

military officials, researchers and corporations.

While some sensitive military data are involved, the computers handling the nation's most sensitive secret information, like that on the control of nuclear weapons, are thought not to have been

"All the News That's Fit to Print"

The New York T

VOL.CXXXVIII . . . No. 47,680 Copyright © 1988 The New York Times NEW YORK, SATURDAY, NOVEMBER 5, 1988

Author of Computer 'Virus' Is Son Of N.S.A. Expert on Data Security

Cornell Graduate Student Described as 'Brilliant'

By JOHN MARKOFF

The "virus" program that has plagued many of the nation's computer networks since Wednesday night was created by a computer science student who is the son of one of the Government's most respected computer security experts.

The program writer, Robert T. Morris Jr., a 23-year-old graduate student at Cornell University whom friends describe as "brilliant," devised the set of computer instructions as an experiment, three sources with detailed knowledge of the case have told The New York Times.

The program was intended to live innocently and undetected in the Arpanet, the Department of Defense computer network in which it was first introduced, and secretly and slowly make copies that would move from computer to computer. But a design error caused it instead to replicate madly out of control, ultimately jamming more than 6,000 computers nationwide in this country's most serious computer "virus" attack.

The dent's program jammed the computers of corporate research centers including the Rand Corporation and SRI International, universities like the University of California at Berkeley and the Massachusetts Institute of Technology as well as military research centers and bases all over the United States.

Meeting with the Authorities

The virus's creator could not be reached for comment yesterday. The sources said the student flew to Washington yesterday and is planning to hire a lawyer and meet with officials of the Defense Communications Agency, in charge of the Arpanet network.

Friends of the student said he did not intend to cause damage. They said he

'VIRUS' ELIMINATED, DEFENSE AIDES SAY

POLAND IS BUYING 3 BOEING AIRLINERS FOR $220 MILLION

EAST BLOC ORDER A FIRST

Sale to Be Financed Through a Lease-Purchase Accord With Western Banks

By AGIS SALPUKAS

The Boeing Company received an order yesterday from the national airline of Poland, the first order for advanced American aircraft from an Eastern bloc country.

The order from the LOT airline is for three 767 wide-bodied aircraft and is worth about $220 million. The transaction is to be financed through a lease-purchase agreement with Western banks, under which the airline will own the planes after 12 years.

Airline officials, at a news conference at the Polish Consulate in New York yesterday, would not identify the

MOSCOW OF ITS A CHARGE

U.S. Expresses Disappointment

President Reagan said yesterday that he was disappointed by the Soviet Union's decision to suspend the withdrawal from Afghanistan. The State Department said the suspension was disturbing.

Marlin Fitzwater, the White House spokesman, said the Soviets' actions "can only increase tensions in the region and raise speculation that they aren't going to live up to the Geneva accords."

But Administration officials nevertheless drew attention to Moscow's statement that the Soviet Union still intends to adhere to the accords, which call for the troop withdrawal to be

Morris worm mechanisms

used vulnerabilities in some versions of:

- mail servers (`sendmail`)
- user information servers (`fingerd`)

also spread using `rsh`/`rexec` (predecessor to ssh)

hid by being called `sh` (default shell)

strings obscured slightly in binary

Eichin and Rochlis, "With Microscope and Tweezers: An Analysis of the Internet Virus of November 1998"

Morris worm intent versus effect

code in viruses tried to avoid "reinfecting" machines

... but not actually effective

Stuxnet

targeted Iranian nuclear enrichment facilities

physically damaged centrifuges

designed to spread via USB sticks

publicly known 2010, deployed 2009

US + Israel gov't developed

according to press reports

Ransomware

encrypt files, hold for "ransom"

decryption key stored only on attacker-controlled server

possibly decrypt files if victim pays

many millions in revenues

accurate numbers are hard to find

Ad injection (1)

internet advertising is big business

... but you need to pay websites to add ads?

how about modifying browser to add/change ads

mostly bundled with legitimate software

Ad injection (2)

5% of Google-accessing clients (2014)

>90% using code from VC-backed firm SuperFish:

$19.3 M in investment (CrunchBase)

$38M in revenue (Forbes, 2015)

defunct after Lenovo root CA incident (2015)

... but founders reported started new, similar venture (JustVisual; according to TechCrunch)

Adware prevalence: Thomas et al, "Ad Injection at Scale: Assessing Deceptive Advertisement Modifications"

Other monetization techniques

obtain banking/etc. passwords

hijacking cameras and blackmail

flood website/services with internet traffic

"cloud" of hijacked machines for computations (e.g. password racking)

Website

linked off Collab

`https://www.cs.virginia.edu/~cr4bd/4630/S2017/`

will include slides, assignments, lecture recordings

Prerequisites

technically CS 2150

CS 3330 will be very helpful

Exams/Assignments

many approx. one week assignments

two midterms — schedule on website

one final

can't make it? need accomodations? tell us ASAP!

Textbook

no required textbook

optional supplementary materials:

Szor, *The Art of Computer Virus Research and Defense*

Smith and Marchesini, *The Craft of System Security*

TAs/Office Hours

posted on website

yes, we will have one

Misc. Policies

possibly exceptional circumstances? ask!

there is a late policy

don't cheat

don't know if it's cheating? ask

On Ethics

don't use someone's computer without their permission

- or in excess of what they've permitted

don't assume it's just a harmless prank

- unintended (but likely) consequences

don't assume the system owner would give you permission

- if you're afraid to ask, it's not okay

On Law

probably illegal (Federal and/or State crime):

accessing computers without authorization

- even if nothing is done with the access

deliberately overloading a service

"backhacking" into a malware operator's machine

deploying a worm that patches security holes

ethics pledge — please read and sign

questions about ethics?

topics outline

prerequisite: assembly review

malware history

cat-and-mouse: anti-malware

software vulnerabilities

- memory management related

Conclusion