Computer and Network Security

Dr Chan Yeob Yeun

Week 7

جامعـــة خليفـــة
Khalifa University

Weekly Lecture Plan

Wk	Contents	Cmt	Wk	Contents	Cmt
1	**Introduction**		9	Foundations of Network Security II	
2	**Foundations of Computer Security**	**Tutorial** **Assig Plan**	10	Network-Based Threats and Attacks	
3	**Identification and Authentication I**		11	Network Security Protocols I	
4	**Identification and Authentication II**	**Quiz 1**	12	Network Security Protocols II	Quiz 3
5	**Access Control**		13	Firewalls	
6	**Modern Computer Attacks**		14	IDS / IPS	Assig Submit
7	**Malicious Code**	**Assig Confirm**	15	Revision and Presentation	
8	Foundations of Network Security I	Quiz 2	16	Exam	

Malicious Software (Malware)

There are a number of categories of malicious software including:

- Viruses
- Worms
- Trojan Horses
- Rootkits
- Backdoors
- Spyware
- Fake AV
- Botnets
- Keystroke loggers
- Logicbombs

Viruses

A Virus is a computer program that is designed to replicate itself by copying itself into other programs stored in a computer.

The ‘virus’ may be benign or may carry a malicious payload that can cause a program to operate incorrectly or corrupt a computer's memory.

A computer virus behaves in a way similar to a biological virus, which spreads by inserting itself into living cells. Extending the analogy, the insertion of the virus into a program is termed an *infection*, and the infected file (or executable code that is not part of a file) is called a *host*.

The first virus is believed to be the Creeper virus that spread through ARPANET (the precursor to the internet) in 1971. It copied itself to remote systems where the message “I’m the creeper, catch me if you can!” was displayed.

Viruses

Many early viruses spread via floppy disks. Users often exchanged data and programs on disks, and machines would boot from floppy disk by default (if present).

Viruses spread by either infecting programs stored on disks or installed them into the disk boot sector to be run when the machine booted. As BBS's became popular, viruses were written to infect popular, often traded software.

Macro viruses started to appear in the 1990's. Macro languages are languages built into software applications (e.g. Microsoft Office). This allows code to be embedded in documents that can run when opened. So if a document that contains a macro virus is transported to different machines, the virus will go with it.

As macros are application based, and not dependent on the operating system, the same macro virus can spread across different systems.

Viruses

To spread itself, a virus must be allowed to execute code and write to memory. Often, viruses will attach themselves to legitimate programs, to be executed when the program is launched.

There are two types of possible behavior when a virus is executed:

Non-resident viruses – on execution they search for and infect other hosts, then transfer control to the application they infected.

Resident viruses – loads itself into memory on execution and then transfers control to the application they infected. It stays active in the background and spreads when possible. When the operating system performs certain actions, the virus will spring into action and start to spread.

Viruses

The possible hosts for a virus include:

- Binary executable files
- Volume Boot Records of hard disks/floppy disks
- Master Boot Records
- General-purpose scripts (batch scripts, shell scripts, VBScripts, etc.)
- System specific autorun script files (e.g. Autorun.inf)
- Documents that contain macros
- Cross-site scripting vulnerabilities in web applications
- Arbitrary computer files (can be exploited by buffer overflow, format string or other)
- Links (do not actually contain virus code, but the linked to site could in scripts, or downloaded executable)

Sophisticated Viruses

As anti-virus systems have developed to detect viruses, viruses have grown more complex and use sophisticated methods to avoid detection.

Viruses may infect files without modifying the size or timestamp (but will still be detected through checksums). Some viruses attempt to disable anti-virus software.

Alternatively, a virus might avoid infecting AV files (or any bait files the AV system creates/monitors) as these will be closely checked.

A virus might also try to intercept any requests made to the operating system and return fake “clean” results.

Most AV predominately rely on signatures, identifying patterns of code in an infected file. To prevent this, viruses can often modify themselves on each infection. More advanced methods use simple encryption to encipher (the bulk of) the code. Polymorphic viruses will mutate while keeping the original algorithm intact, so the code will look different, but the effect remains the same.

Worms

A worm is a program that actively transmits itself over a network to infect other computers. It may also carry a payload.

The First worm: On November 2, 1988, Robert T. Morris (Jr.), a Cornell University graduate student, unleashed a worm onto the Internet from MIT that infected between 6,000 and 9,000 computers, a significant portion at the time. He was a computer science student and claimed that he was interested in determining how far and how quickly the worm could spread throughout the network, but he did not anticipate that it would cause as much trouble as it did due to his own coding error on the program's logic. He was convicted and sentenced to three years of probation and 400 hours of community service, in addition to a $10,000 fine. The program took advantage of a hole in the debug mode of the Unix *sendmail* program, which runs on a system and waits for other systems to connect to it and give it email, and a hole in the finger daemon *fingerd*, which serves finger requests.

Worms

Whereas viruses tend to cause damage to the target computer, worms do more damage to the network, consuming bandwidth and denial of service.

The payload may do more than just spread the worm. It may:

- Delete files
- Encrypt files (for extortion)
- Send documents via email
- Install a backdoor/create a “bot”
- Send spam
- Launch a DoS

Worms exploit vulnerabilities in systems to propagate, so can be defended against by applying regular security updates/patches. End users need also be careful as worms can spread through email.

Trojans

“A Trojan is a harmful piece of software that looks legitimate. Users are typically tricked into loading and executing it on their systems.”

There is no limit to what type of program a Trojan can appear to be. Malware authors can go to great lengths to create apps to get their code on other users, or even inject it into someone else’s if it isn’t hosted securely.

Trojans are typically smaller applications: screensavers, simple hacking tools, add-ons, cracks to run popular games without the disks, anything that a user would not get at a reputable website (and needs to go to an untrusted one).

Unlike viruses or worms, self-propagation is not a features of Trojans. However, due to their exploitation of peoples desires for free software, they are just as widespread (sometimes more so).

Keystroke loggers (Keyloggers)

A keylogger is a type of surveillance software (considered to be either software or hardware) that has the capability to record every keystroke you make to a log file, usually encrypted.

A keylogger can record instant messages, e-mail, and any information you type at any time using your keyboard. The log file created by the keylogger can then be sent to a specified receiver. Some keylogger programs will also record any e-mail addresses you use and Web site URLs you visit.

Keyloggers, as a surveillance tool, are often used by employers to ensure employees use work computers for business purposes only. Unfortunately, keyloggers can also be embedded in spyware allowing your information to be transmitted to an unknown third party.

The most dangerous aspect of keyloggers is that they can capture information before it is encrypted, defeating any number of protection methods.

Hardware Keylogger

Independent hardware devices attached to the machine to be monitored. Hard to detect unless they are specifically looked for (visually, no scans would reveal them). Usually require the attacker to regain access to get the data, but more sophisticated devices will be able to broadcast it.

- KeyLlama 4MB PS/2 KeyLogger - Price: $60.88
- KeyLlama 4MB USB KeyLogger - Price: $92.88

Use 64-bit encryption

http://keyllama.com/

Software keyloggers

More widely used as they can be installed remotely as part of an attack. Data is emailed out from the monitored machine regularly. Software keyloggers are not limited by physical memory allocations and so can store more information.

- Blazing Tools Perfect Keylogger Lite
- Keyboard Spectator Free
- Ardamax Keylogger Lite
- Quick Free Keylogger
- Shadow Keylogger (Portable) 1.1.0 freeware

Spyware

Spyware is a type of malware that is installed on computers and collects information about users without their knowledge. Typically, spyware is secretly installed on the user's personal computer. Sometimes, however, spyware, such as keyloggers, are installed by the owner of a shared, corporate, or public computer on purpose in order to monitor other users. Spyware can monitor any of the following:

- Keystrokes
- Email messages
- IM chat sessions
- Websites visited
- Applications opened
- Screenshots
- Clipboard

Spyware

Spyware generally gets installed through misleading the user or by exploiting vulnerabilities.

The user might be presented with a useful utility (e.g. “web accelerator” or helpful software agent), which fails to mention what happens in the background. It may come coupled with a desirable piece of software (many p2p programs come bundled with an offer of spyware). Or it might be delivered via a drive-by downloads/installs.

Spyware is most often financially motivated. The monitoring of internet habits can be used to provide targeted advertisements, ads that are tailored to the individual and so more likely to be of interest. These are usually in the form of pop-up windows.

Spyware

While the term *spyware* suggests software that secretly monitors the user's computing, the functions of spyware extend well beyond simple monitoring. Spyware programs can collect various types of personal information, such as Internet surfing habits and sites that have been visited, but can also interfere with user control of the computer in other ways, such as installing additional software and redirecting Web browser activity.

Spyware is known to change computer settings, resulting in slow connection speeds, different home pages, and/or loss of Internet or functionality of other programs.

Where multiple infections occur, the machine is likely to suffer severe instability as the different spyware programs conflict. Some spyware even disables security settings to allow further installations.

Spyware

Legitimate spyware:

- Stealth Voice Recorder
  - http://www.topofbestsoft.com/
- Digi-Watcher Video Surveillance - with motion detection
  - http://www.digi-watcher.com/
- Stealth Website Logger
  - http://www.amplusnet.com/
- Desktop Spy Screen Capture Program
  - http://www.canadiancontent.net/
- Print Monitor Spy Tool
  - http://spyarsenal.com/

Spyware

Illegitimate spyware:

- Internet Optimizer (aka DyFuCA) - redirects broken links/error pages to advertising pages
- Movieland - Subscription-based media service, once trial is installed, will bombard the user with pop-ups and music that cannot be easily uninstalled. Demands payment of $29.95 (Ransomware)
- CoolWebSearch - Uses a drive-by installation. Changes infected computer’s homepage, creates pop-up ads, inserts links on random text leading to ad websites, pay-per-click search engines, adds links to gambling and pornography websites.
- Zlob - Masquerades as a needed video codec. Redirects the default internet search and home pages, attempts to download and execute malicious software.

Fake AV

Illegitimate AV:

- Programs that report pop-up warnings and alarms
  - Windows Security alert
  - Windows reports that computer is infected.
  - Antivirus software helps to protect your computer against viruses and other security threats.
  - Click here for the scan your computer.
  - Your system might be at risk now.
- Users are then directed to download AV software where they have to pay. The downloaded AV software typically contains more trojans or does nothing.
- Recent fake anti-virus sites have even been imitating the look and feel of the Windows user interface.
- Big business these days
  - 15% of all malware these days are Fake AV
  - Generates more revenue than legitimate AV

Rootkits

A *rootkit* is a piece of malicious code that goes to great lengths not to be discovered or, if discovered and removed, to reestablish itself wherever possible.

Installation requires the attacker to have root or administrator access. Once installed, the rootkit maintains this access over time and actively hides its presence from other processes.

Typically a rootkit will come with some malicious payload (keyloggers, sniffers, bot, spam delivery) and/or a backdoor which will also remain concealed. Rootkits are hard to detect with common antivirus programs.

There are (a small minority of) examples of useful rootkits:

- Honeypots
- Anti-virus (Kaspersky)

Rootkits

A rootkit generally works as follows: Whenever a user executes a command that would show the presence of the rootkit, the rootkit intercepts the call and filters the results. Interception is done by concepts of hooking! Rootkits can be implemented in a variety of different ways:

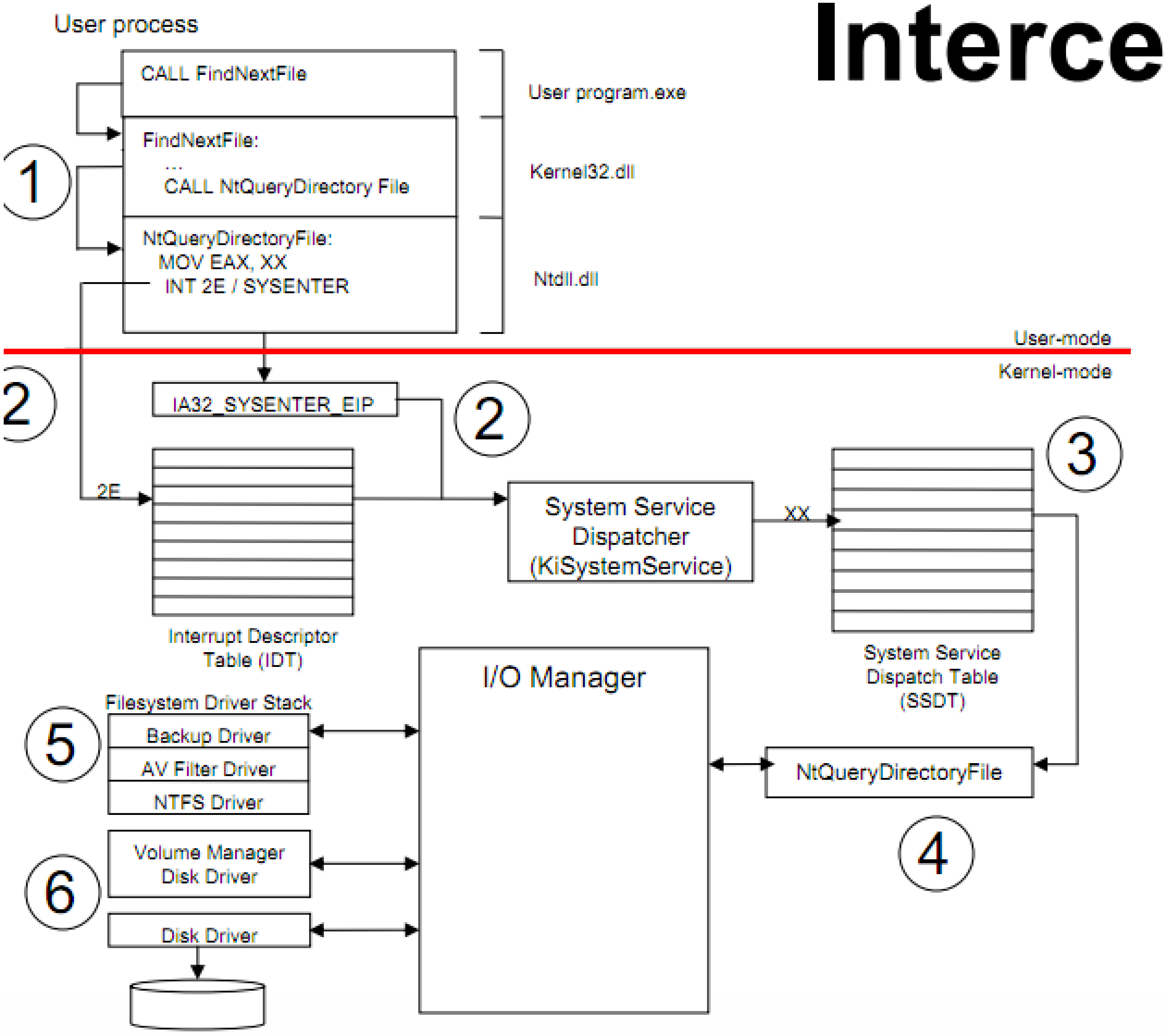
Application level: The rootkit replaces or modifies existing application binaries using hooks, patches, injected code or other means

Library level: Modifies the system libraries (DLLs in Windows) to hide its presence

Kernel level: The rootkit adds/replaces code in the kernel or device drivers. Gives unrestricted access, and very difficult to detect. Can impact system stability if not well written.

Recent research has been done on rootkits at even lower levels (hypervisor, boot loader, firmware, hardware).

Interception



1. User-mode hooks
2. IDT / SYSENTER hooks
3. SSDT hooks
4. Kernel code patching
5. Layered driver
6. Driver hooks

Slide by Chris Ries, VigilantMinds Inc, on “Windows Rootkits”

Rootkits

Detection:

The problem with detecting an installed rootkit is that the operating system has been fundamentally compromised and cannot be trusted. Rootkit detectors on live systems are usually limited to being able to find a rootkit before it is installed. The most reliable method is to boot from a trusted medium (live CD, USB drive), and safely load the suspected drive.

Security vendors have attempted to add rootkit detection abilities to their solutions, which has lead to a familiar struggle (counter-measure, counter-counter-measure).

Prevention can be achieved by calculating hashes of all system files after a clean install and making a comparison at a later date to see what changes have been made.

Cross-view comparison is a more effective mechanism to detect rootkits.

Backdoors

A **backdoor** (or trapdoor) in a computer system (or a cryptosystem, or even in an algorithm) is a method of bypassing normal authentication or obtaining remote access to a computer, while intended to remain hidden to casual inspection.

The backdoor may take the form of an installed program, a modification to a legitimate program or simply a leftover artifact from development.

Once a system has been compromised, one or more backdoors may be installed in order to allow easier access in the future.

Backdoors may also be installed prior to malicious software, to allow attackers entry.

Backdoors

Example: Back Orifice (BO)

This is a remote system administration tool or malicious backdoor depending on your point of view. It was released in 1998 by the CULT OF THE DEAD COW, to illustrate the lack of security in Windows 98. A server is installed on one computer (does not require user intervention so can be delivered by a Trojan), which allows that machine to be controlled remotely. The server can hide from simple scans by the user, and can:

steal passwords

format hard disk

delete files

execute commands

BO can be detected by most antivirus programs.

Botnets

Botnet is a term for a collection of software robots, or bots, that run autonomously and automatically.

While the term "botnet" can be used to refer to any group of bots, such as IRC bots, this word is generally used to refer to a collection of compromised computers (called Zombie computers) running software, usually installed via drive-by downloads exploiting Web browser vulnerabilities, worms, Trojan horses, or backdoors, under a common command-and-control infrastructure.

Botnets

The creator of the botnet, called the bot herder or bot master, controls the collection of machines remotely. Commands will be sent to one or more C&C servers and propagated through the botnet.

IRC is often used to control botnets because it is flexible, easy to use and because public servers can be used as a communication medium. Simple tricks can be done by attackers to cover their identity (anonymous proxies or IP address spoofing).

Botnets are typically sold and used for such illegal activities as DDoS, spam, click fraud, etc. The number of machines in a botnet tends to be in the thousands, but examples with millions of infected machines have been found[1].

[1] http://www.spywared.com/security-news/2009/07/28/top-10-us-botnet-threats.html

Sniffer

A sniffer is a program and/or device that monitors data travelling over a network. This is a network analyzing tool that has both constructive and destructive uses. There are many different tools available that vary from simple command line interface to complicated GUI with statistics presented in a variety of graphs/animations.

In an Ethernet environment, all machines receive packets meant for one machine. If a packet is not meant for a machine they are supposed to discard it. A sniffer will ignore this and can read all traffic passively. A switched network is harder to sniff as the switch will intelligently route packets, but it is still possible (using ARP spoofing or MAC flooding, which are active sniffing methods).

The following protocols are vulnerable to sniffing (in that passwords and data can be sent in the clear: HTTP, SNMP, NNTP, POP, IMAP, FTP.

Logic bombs

Logic Bombs

Logic Bombs are pieces of code that are deliberately inserted into code in order to cause damage to a system.

They are typically created by disgruntled/former employees.

They are not considered viruses because they do not replicate. They are not even programs in their own right but rather camouflaged segments of other programs.

The fact that they are written by developers who have complete access and in-depth knowledge of the system means that they are very difficult to detect.

Logic bombs

Examples of Logic Bombs

Date	Bomber	Victim	Payload	Sentence
2006	Roger Duronio	UBS	Damage the company’s computer network (to reduce the stock value)	8 years, $3.1 million
2005	Jeffery Gibson	St. Cloud Hospital	Disabled the software he developed	5 years probation, $28,000, community service
2003	Yung-Hsun Lin	Medco Health Solutions	Wiped out data on 70 servers	30 months, $81,200
1992	Michael Lauffenburger	General Dynamics	Delete vital project data (so he could return as a highly paid consultant to fix the problem)	$5,000

Malware Detection

Malware is a general term that covers a number of different classes of code. Within each class, there is a large number of specimens, which is constantly growing. The techniques to detect and remove malware have struggled to keep up with the evolution of malware.

There are good general malware detection techniques, and several tools specific to each class of malware.

But the tools are not 100% effective at detecting every threat, and complete removal is not always possible.

The best defense is to prevent rather than to detect and clean up an infection.

Malware Detection

Viruses have been around long enough to be known by most computer users. While laws exist in most counties to ban virus writing, the cost of catching and prosecuting means most authors go unpunished.

It does not appear that the constant flow of new virus is going to be stemmed any time soon, so prevention and detection methods will be needed for the foreseeable future.

General protection techniques:

1. Restrict your file downloading to known or secure sources
2. Do not open any e-mail attachment that was not expected
3. Use an up-to-date anti-virus program or service
4. Enable macro protection in applications
5. Create backup copies of all your important data

Malware Detection

Anti-virus software is software used to prevent, detect and remove malware, including viruses, worms, Trojans, spyware, etc. There are two basic techniques used:

1. Signature Scanning:

AV programs have a database of unique sequences of binary code that identify malware (code signature). All programs will scan for these signatures. Depending on the program and type of scan requested it might only scan certain files (OS files, libraries, executables), or it might scan the entire drive, opening archives and compressed files. The sheer number of viruses in the wild (and their ability to permute themselves) means that it is not feasible for an anti-virus program to search for all known viruses.

Malware Detection

New signatures are obtained by user reporting, research analysis and web searching. To date, there are 1.6 M signatures!

Even ignoring the problem with numbers, there is still a fundamental problem with a signature approach, that it is reactive. There is a window of vulnerability the users are exposed to. The exploit cycle works as follows. A vulnerability becomes publicly known, a patch is developed, the patch is released, the patch is deployed.

This cycle can take days, weeks or even months, depending on the priorities of the developer. And patches are not always applied. It is not uncommon for vulnerabilities to be exploited long after the patch has become available.

The vulnerability can be exploited at any point up until deployment of the patch, even before the vulnerability is publicly known (zero day attack).

Malware Detection

Virus writers have adapted to signature scanning by creating polymorphic viruses that change or encrypt themselves to avoid detection.

2. Heuristic Scanning:

The more sophisticated approach is to look for virus-like behaviour, not often found in legitimate software programs. For example, a program will rarely edit the Windows Registry, which is something viruses often do. This method looks for malicious intent rather than a definite fingerprint. On finding a suspicious file, the scanner may take different actions: advise the user, quarantine the file, send the file to the AV company for further analysis.

Malware Detection

Here are some common approaches to heuristic scanning:

Content Filtering compares the scanned code with a built-in rule base that classifies suspect behavior. This might include such actions as modifying .exe files, making changes to the registry, changing personal settings. This does not identify a specific virus, but looks for likely ways that a virus writer might code the mechanism for propagation or payload delivery.

Sandboxing allows incoming code to run inside a simulation of the computing environment. The code cannot harm or infect the rest of the system. The sandboxing software looks for specific activities and will alert the user if found, or run the program normally otherwise. Sandboxing can defeat code obfuscation techniques as the virus will decode itself.

Malware Detection

Behavior Analysis looks at the overall system performance. It tracks everything the system does and monitors for any deviation from standard operating procedures.

Application Whitelist uses a database of known good applications, rather than try to classify all bad ones. Examples: Malwarebytes Anti-Malware, Bit9 Parity and Savant Protection.

The main advantage of heuristic scanning is to potentially identify new threats. The disadvantage is an inexact science, with many false positives and false negatives. This can be seen in independent comparisons and evaluations of AV programs.

Malware Detection

Anti-Virus Comparative August 2010 - On-demand detection

Name	Detection %	FP	MB/s	Award
G DATA	99.9	15	10.3	Advanced+
AVIRA	99.8	10	15.1	Advanced+
Avast	99.3	9	17.2	Advanced+
Bitdefender	99.3	4	12.1	Advanced+
F-Secure	99.2	2	11.9	Advanced+
eScan	99.2	5	8.9	Advanced+
Symantec	98.7	9	13.3	Advanced+
ESET	98.6	6	9.6	Advanced+
PC Tools	98.1	7	9.7	Advanced+
Trustport	99.8	19	9.4	Advanced
McAfee	99.4	24	12.3	Advanced
Panda	99.2	98	14.6	Advanced
AVG	98.3	19	13	Advanced
Kaspersky	98.3	46	9.8	Advanced
Microsoft	97.6	3	5.9	Advanced
Sophos	96.8	13	10.6	Advanced
Norman	96.6	98	4.4	Standard
K7	96.6	50	11.7	Standard
Trend Micro	90.3	23	12.8	Tested
Kingsoft	80.1	45	11.6	Tested

Malware set were obtained from honeypots, vendors, malware downloaders and infected websites. Samples were frozen on the 6th of August 2010. Products were updated and frozen on the 16th of August 2010. Nearly 0.9 million samples were used, made up of:
68.3% Trojans
13.5% Backdoors/Bots
13.4% Worms
12.3% Windows viruses
0.7% Scripts and macro viruses
1.8% Other malware

http://www.av-comparatives.org/comparativesreviews/main-tests

Malware Detection

Anti-Virus Comparative February 2010 - Proactive test

Name	Detection %	FP	Award
TrustPort	63	4-15	Advanced+
G DATA	61	4-15	Advanced+
Kaspersky	59	4-15	Advanced+
Microsoft	59	0-3	Advanced+
AVIRA	53	4-15	Advanced+
ESET NOD 32	52	0-3	Advanced+
F-Secure	52	0-3	Advanced+
BitDefender	50	0-3	Advanced+
eScan	50	0-3	Advanced+
Panda	63	15+	Advanced
K7	50	15+	Advanced
Symantec	43	4-15	Advanced
AVG	34	4-15	Advanced
Sophos	32	4-15	Advanced
Avast	29	4-15	Advanced
McAfee	38	15+	Standard
Norman	27	15+	Standard
Trend Micro	26	15+	Standard
PC Tools	17	4-15	Standard
Kingsoft	11	15+	Tested

The Proactive test froze the products on the 10th of February, and tested them against new malware discovered between the 11th and 18th of February.

The new samples consisted of:
27,271 samples:
24% Worms
13% Backdoors
62% Trojans
1% Other malware

http://www.av-comparatives.org/comparativesreviews/main-tests

Malware Detection

Integrity Checking: An alternative to scanning is integrity checking. A snapshot of all the program files is taken initially, and at regular intervals this is re-evaluated. The user is notified of any changes to the files. Cryptographic hashes are used as well as time and size details. Tripwire[1] is probably the best know system integrity verifier.

This is useful against file infector viruses, but less so against macro/script viruses that use documents that are expected to change regularly.

Scheduling: Most AV programs will schedule regular scans of the system. This will find existing infections. Much more important is real-time scanning, stopping viruses before they have a chance to execute on the local machine. This can include e-mail scanning, download scanning, script scanning, etc.

[1] http://www.tripwire.com/

Malware Detection

Pricing model: There are free anti-virus programs (AVG, Avira, Avast), but commercial products tend to have more comprehensive features and more accuracy. Paid programs usually have a subscription cost for regular updates.

Online Antivirus Services: Some companies offer on-demand online services for malware detection. These range from individual file scanner (Kaspersky File Scanner[1]), to complete scan from the browser (BitDefender Online Scanner[2]), to thin-client complete scan and removal (Trend Micro’s HouseCall[3]). These are generally free services, providing the security of their technology for free, and try to sell the convenience of the full product.

Other important criteria include: ease of use, compatibility, GUI, language, additional protections (URL filter, HIPS), support, etc.

[1] http://www.kaspersky.com/scanforvirus

[2] http://www.bitdefender.com/scanner/online/free.html

[3] http://us.trendmicro.com/us/housecall/

Malware Detection - Top 5 Malware

The current most prevalent pieces of malware are:

	Name	# Infections
1	Net-Worm.Win32.Kido.ir	371564
2	Virus.Win32.Sality.aa	166100
3	Net-Worm.Win32.Kido.ih	150399
4	Trojan.JS.Agent.bhr	95226
5	Exploit.JS.Agent.bab	81681

The Conficker/Kido worm uses several infection vectors: Autorun, NetBiOS, Dictionary attack on admin accounts, Windows exploits. Deploys spam bot and scareware. The Sality/Daprosy Worm spreads via LAN connections, USB devices and email (it actually disguises itself as an anti-virus product). It is known to log keystrokes to capture sensitive information.

Courtesy of Kaspersky
http://www.securelist.com/en/analysis/204792141/Monthly_Malware_Statistics_September_2010