Internet Security

by

Alan S H Lam

Internet Security

I. Aware of the Risks

- The threats

II. How they hack in

- Two real case studies with live demo

III. Fighting back

- Counter measures and strategies

IV. Q&A and discussion

Part I Aware of the risks

The Threats

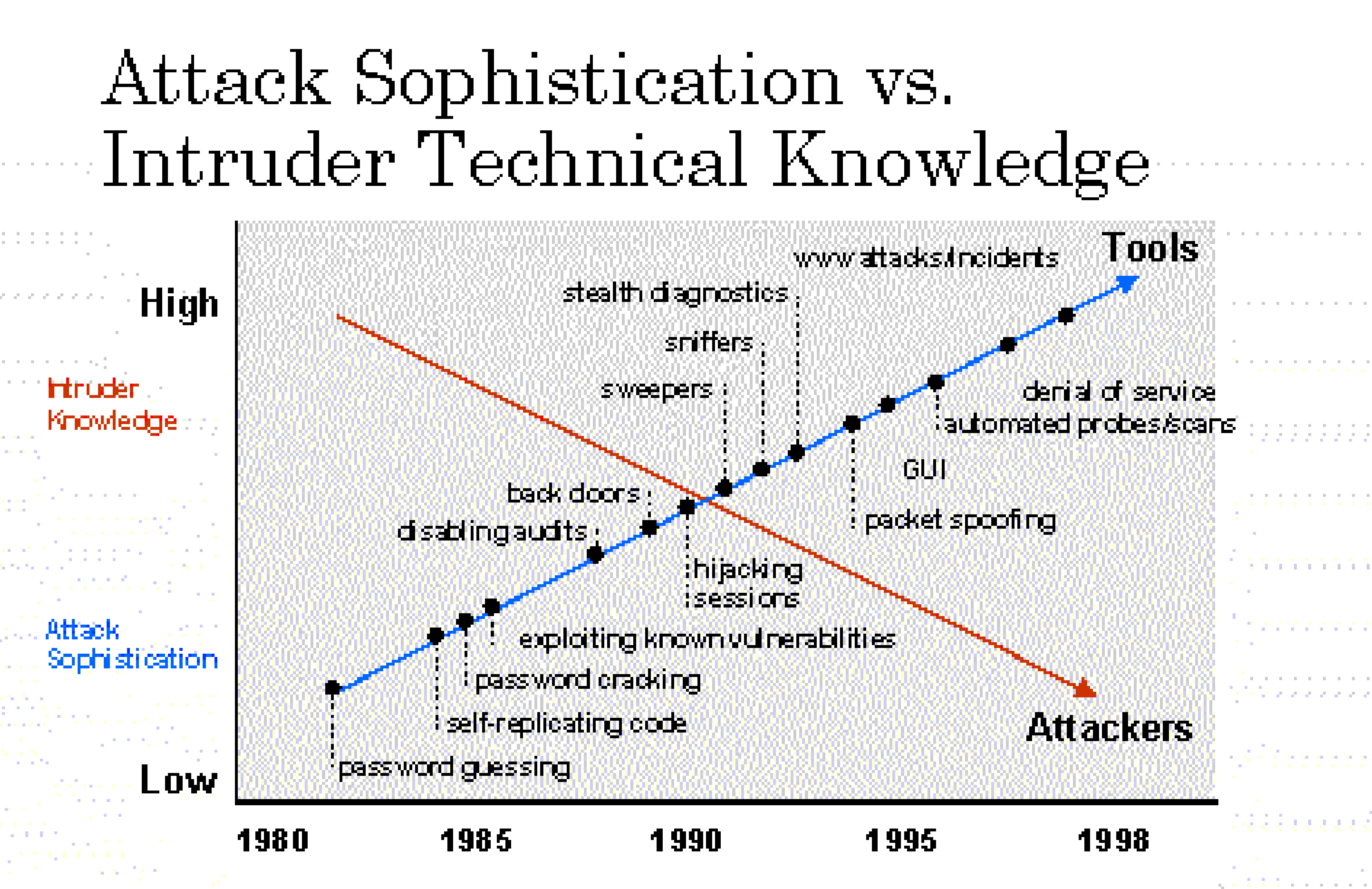
The Threats

- Hacker Technologies
  - Internet Engineering
  - System Administration
  - Network Management
  - Reverse Engineering
  - Distributing Computing
  - Cryptography
  - Social Engineering

The Threats

- Hacking Tools become more and more sophisticated and powerful in term of
  - Efficiency
  - Distributing
  - Stealth
  - Automation
  - User friendliness

The Threats


Attack Sophistication vs.
Intruder Technical Knowledge
High
Low
Intruder Knowledge
Attack Sophistication
Tools
Attackers
www attacks/incidents
stealth diagnostics
sniffers
sweepers
denial of service
automated probes/scans
GUI
packet spoofing
back doors
disabling audits
hijacking sessions
exploiting known vulnerabilities
password cracking
self-replicating code
password guessing
1980
1985
1990
1995
1998

The Threats

- These hacking tools could be easily download from the Internet =>
  - Hacker tool ability increases
  - Knowledge of hacker decreases
  - Population of hacker increases
  - Some day, even elementary school kid may hack into your system

The Threats

- Your host does not need to be as famous as yahoo or ebay to be targeted
    - They need a place to hide their trace
    - They need your host as a stepping stone to hack other sites
    - They need your host resource to carry out their activities

The Threats

- Your host security weakness can be identified by scan tool
- Security of any network on the Internet depends on the security of every other networks
- No network is really secure

The Threats

- The trends
  - Hacking activities become more and more common
  - Poor management networks will become the hackers playground

The Threats

- The Trends
  - From Jan to April 2000, our site has received the following security warning
    - Web page defacement
    - Unauthorized system access
    - Port scanning
    - Ping broadcast scanning
    - Telnet probe scanning

Part II How They Hack In

Two real case studies

How they hack in

- General Steps
  - Locate the victim host by some scanning program
  - Identify the victim host vulnerability
  - Attack the victim host via this vulnerability
  - Establish backdoors for later access

How they hack in

Some hacking tools can automate the above steps into a single command.

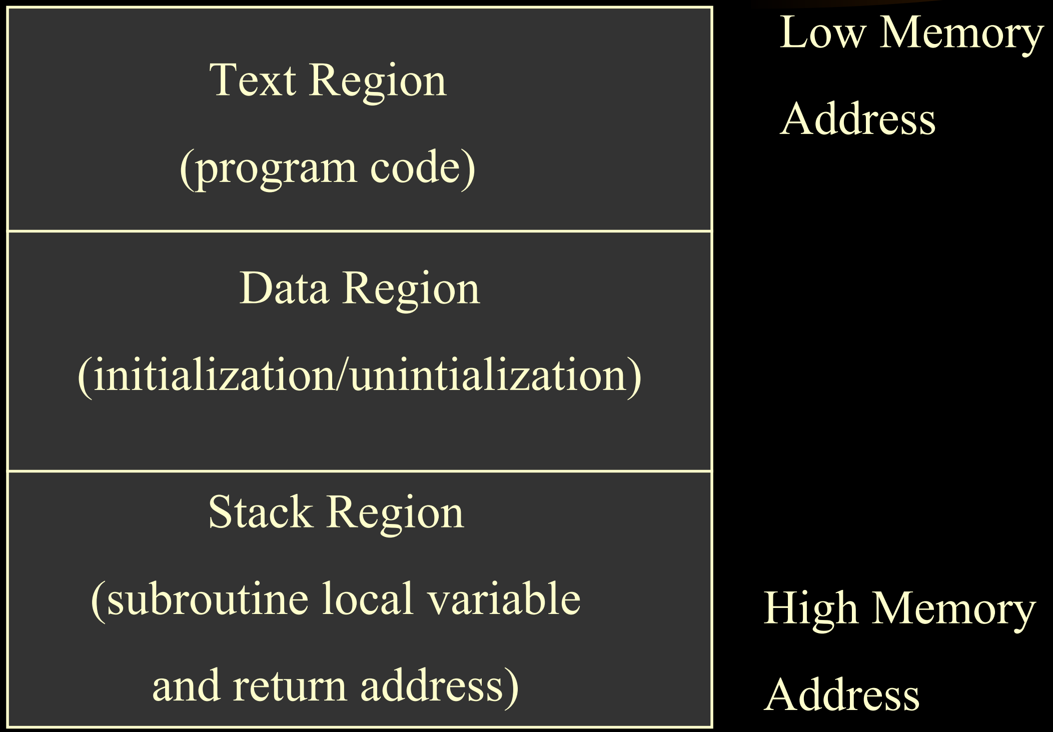
- After break-in, use this victim host to
  - hack other network
  - use this victim host resource to carry out their activities
  - Web page defacement for certain assertion

How they hack in

- Buffer Overflow Exploit
  - stuffing more data into a buffer than it can handle
  - it overwrites the return address of a function
  - it switches the execution flow to the hacker code

How they hack in

- Buffer Overflow Exploit



Text Region

(program code)

Data Region

(initialization/unintialization)

Stack Region

(subroutine local variable

and return address)

Low Memory

Address

High Memory

Address

Process Memory Region

How they hack in

- Buffer Overflow Exploit

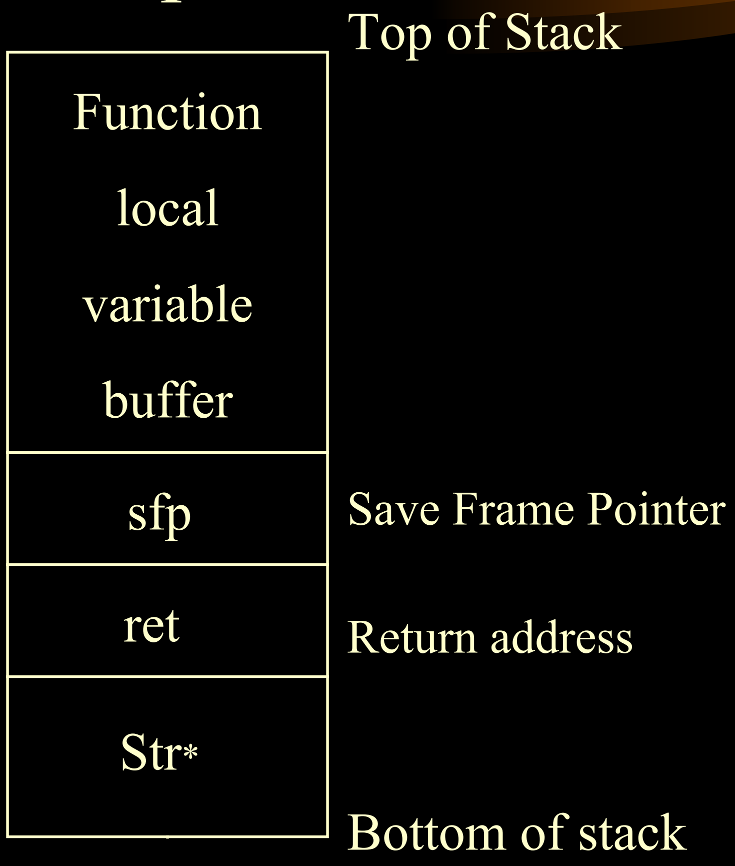
```
void function(char *str) {
   char buffer[16];

   strcpy(buffer,str);
}

void main() {
  char large_string[256];
  int i;

  for( i = 0; i < 255; i++)
    large_string[i] = 'A';

  function(large_string);
}
```



How they hack in

- Real Case Study I
  - Hackers first located the victim hosts by sunrpc scan of 137.189 network
  - Break-in the victim hosts via amd (Berkeley Automounter Daemon) buffer overflow vulnerability
  - Created backdoor on port 2222 by starting a second instance of inetd daemon
  - Used the victim hosts to scan other networks

How they hack in

- Real Case Study II
  - Hackers first located the victim hosts by BIND port 53 scanning
  - Identify the victim OS (a telnet probe)
  - Set up a trap DNS daemon at the hacker DNS server
  - Kicked the victim hosts to query the hacker DNS server
  - Break-in victim hosts via BIND buffer overflow
  - Established back door accounts at the victim hosts
  - Distribute, built and operated the IRC Bot (eggdrop)

Part III Fighting Back

- Get Your Security Profile
- Set Your Security Policy
- Build your Firewall and IDS

Get Your Security Profile

- Act as a hacker and try to break-in your host
  - Port scan your host and see what network ports are open
  - Figure out if the version of your host OS and software applications are vulnerable
  - Can you cover up your trace after break-in? (Does your host have any monitoring or intrusion detection system)
  - Can you easily establish back door after break-ins? (Have you built any firewall?)

Set Your Security Policy

- There is always a trade off between security and convenience
- Identify your host services
  - shutdown any unnecessary ports and build the kernel as minimum as possible
- Identify your target users, trusted hosts and networks so that you can formulate your host access lists
- Set up your firewall
  - use private IP network
  - use proxy servers

Set Your Security Policy

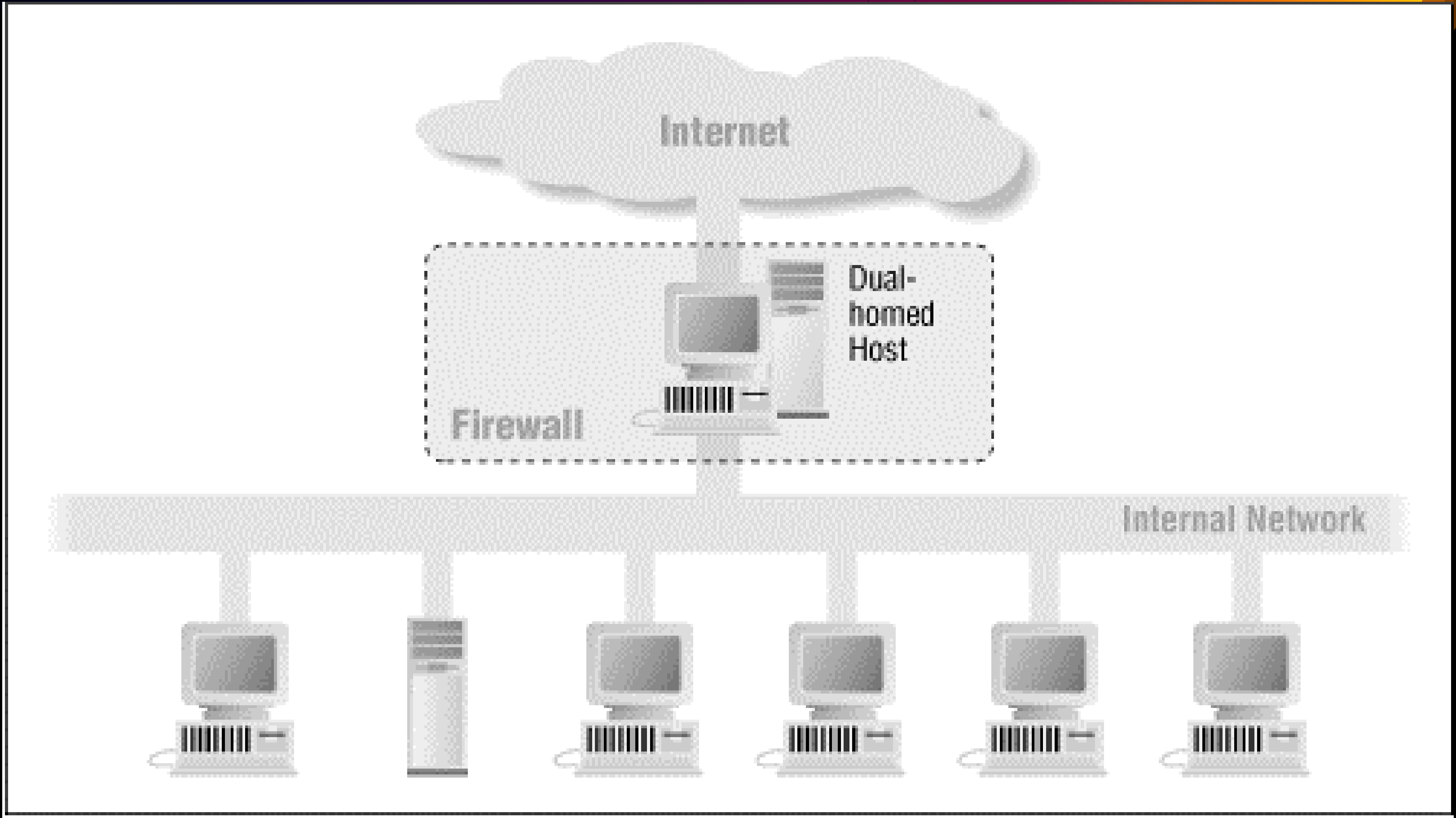
- Set up your monitoring and intrusion detection systems
  - COPS, tripewire, tcpdump, snmp
- Set up you operation codes/rules such as
  - read only file system mounting
  - ssh login
  - sudo
  - restrict login shell
- Set up your recovery plan
  - recovery procedure and backup scheme

Build your Firewall and IDS

- Control and monitor the traffic IN and OUT of your network
- Block any unnecessary network connection from non-trusted hosts and networks
- Define your access rules according to your security policy
- Use packet filtering and Application Proxy
- Build sniffer to monitor your internal network traffic

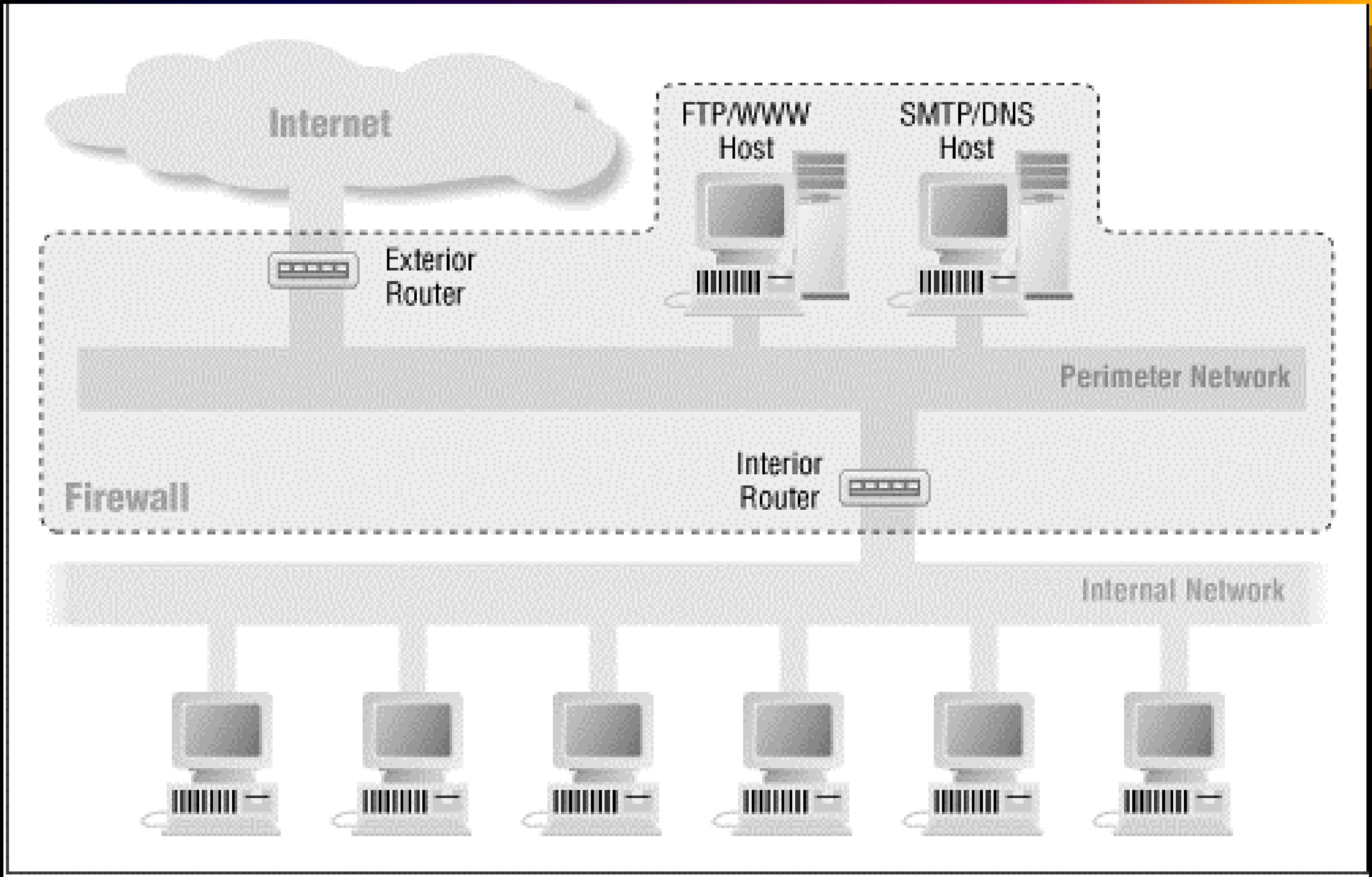
Firewall Architecture

- Dual-home host architecture



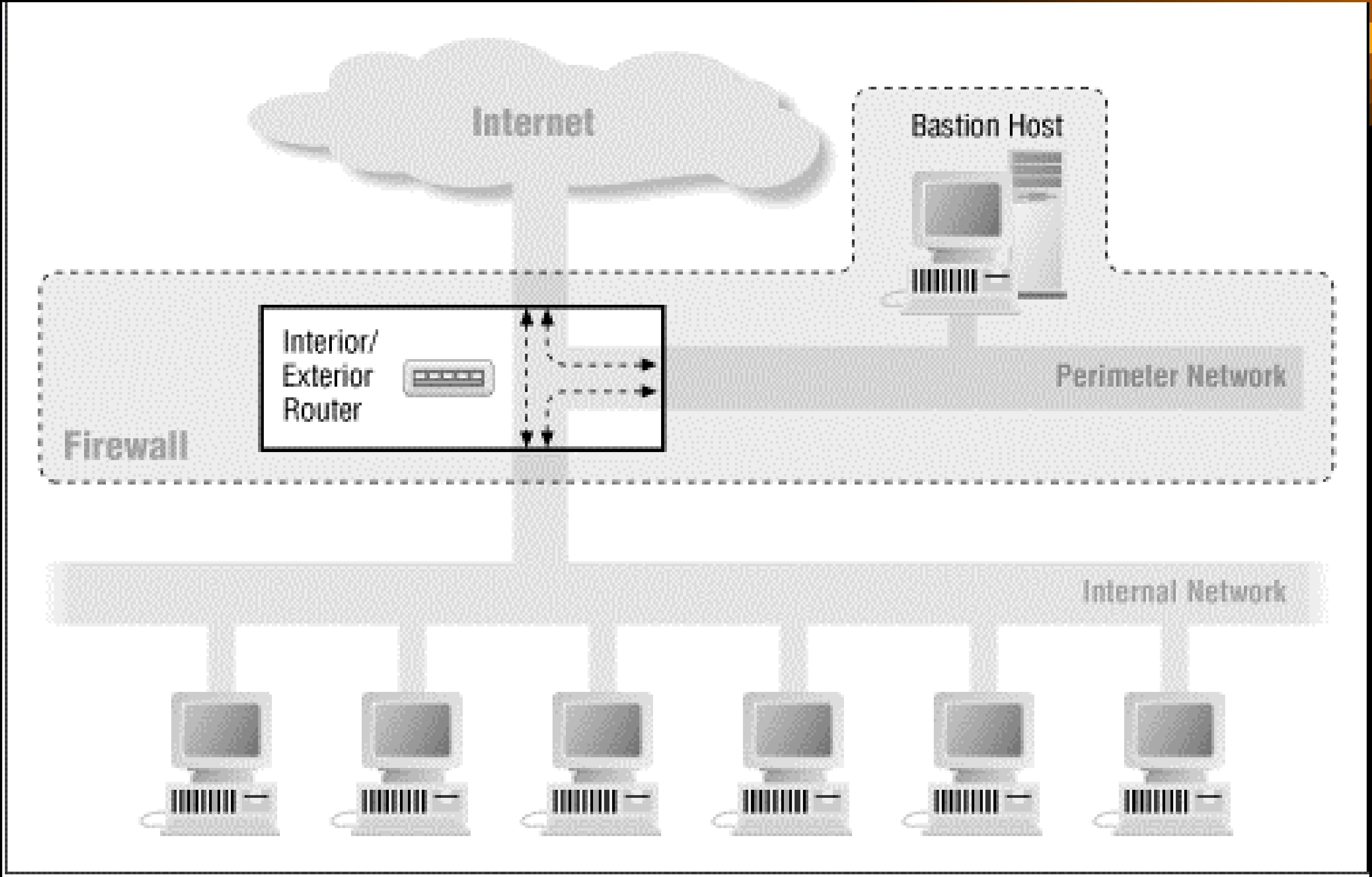
Firewall Architecture

- Architecture using two routers



Firewall Architecture

- Architecture using a merged interior and exterior router



Build Your Firewall

How it protects your network

- prevent port scanning
- prevent DDOS attack and IP spoofing from your host
- block any unnecessary network port opening
- increase the difficulty of creating back door after break-in
- facilitate the network monitoring and network intrusion detection

References

- Attack Sophistication VS Intruder Technical Knowledge
  - http://www.cert.org/sepg99/sld010.htm
- Systems Compromised Through a Vulnerability in am-utils
  - http://www.cert.org/incident_notes/IN-99-05.html
- CERT Advisory CA-99-12 Buffer Overflow in amd
  - http://www.cert.org/advisories/CA-99-12-amd.html
- Real Case Study I (Buffer Overflow in amd)
  - http://home.ie.cuhk.edu.hk/~shlam/ed/hack/case1

References

- CERT Advisory CA-99-14 Multiple Vulnerabilities in BIND
  - http://www.cert.org/advisories/CA-99-14-bind.html
- Real Case Study II (Vulnerabilities in BIND)
  - http://home.ie.cuhk.edu.hk/~shlam/ed/hack/case2
- Scans and Probes
  - http://www.cert.org/current/current_activity.html#scans
- Building Internet Firewall
  - By Chapman & Zwicky, O'Reilly ISBN 1-56592-124-0