



X'CON 2003

《大规模网络中蠕虫主动防治技术研究》
--利用DNS服务抑制蠕虫传播

作者：郑辉

日期：2003.12.22



X'CON 2003

大规模网络中蠕虫主动防治技术研究

--利用DNS服务抑制蠕虫传播

郑 辉

教育科研网应急响应组

zhenghui@ccert.edu.cn



内容



X'CON 2003

n 为什么选择**DNS**服务

n 利用**DNS**服务的方法

n 系统整体框架设计

n 基于配置视图方式的系统实施方案

n 基于端口转发方式的系统实施方案

n 性能分析、实施效果



为什么选择DNS服务



X'CON 2003

- n 大部分Internet应用都会用到DNS服务;
- n 加快染毒用户响应速度;
- n 可以引导用户到指定机器, 相较于其他方式增强了交互性;
- n 减少用户和网管人员的正面冲突;
- n 减少网管人员工作量;
- n 疏导方式对网络流量的影响更小;
- n 实施时间短, 见效快;

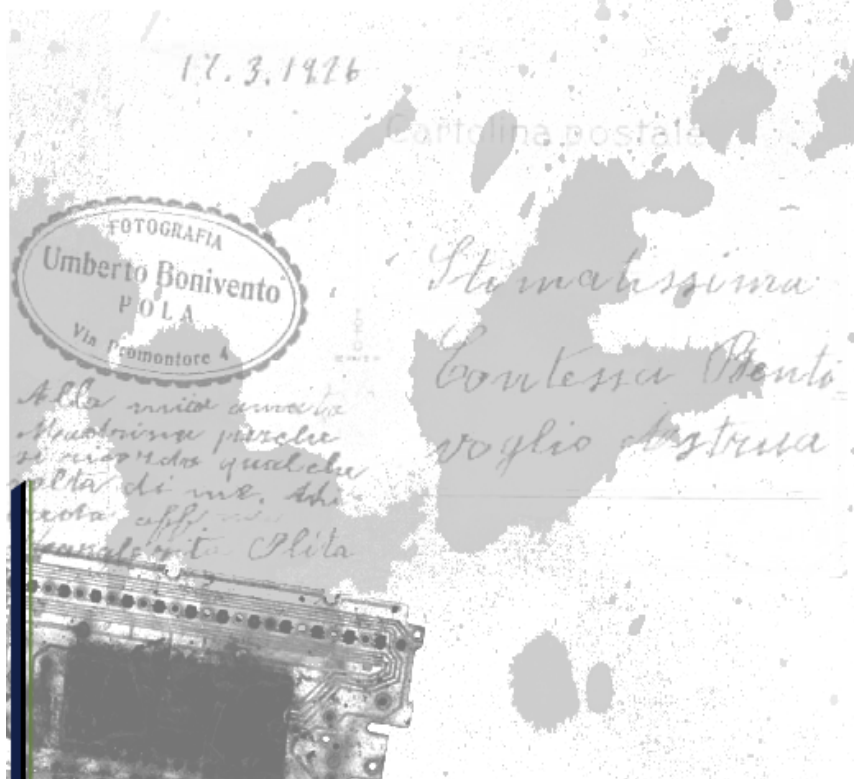
利用DNS服务的方法



X'CON 2003

n 配置视图

n 端口转发



配置视图



X'CON 2003

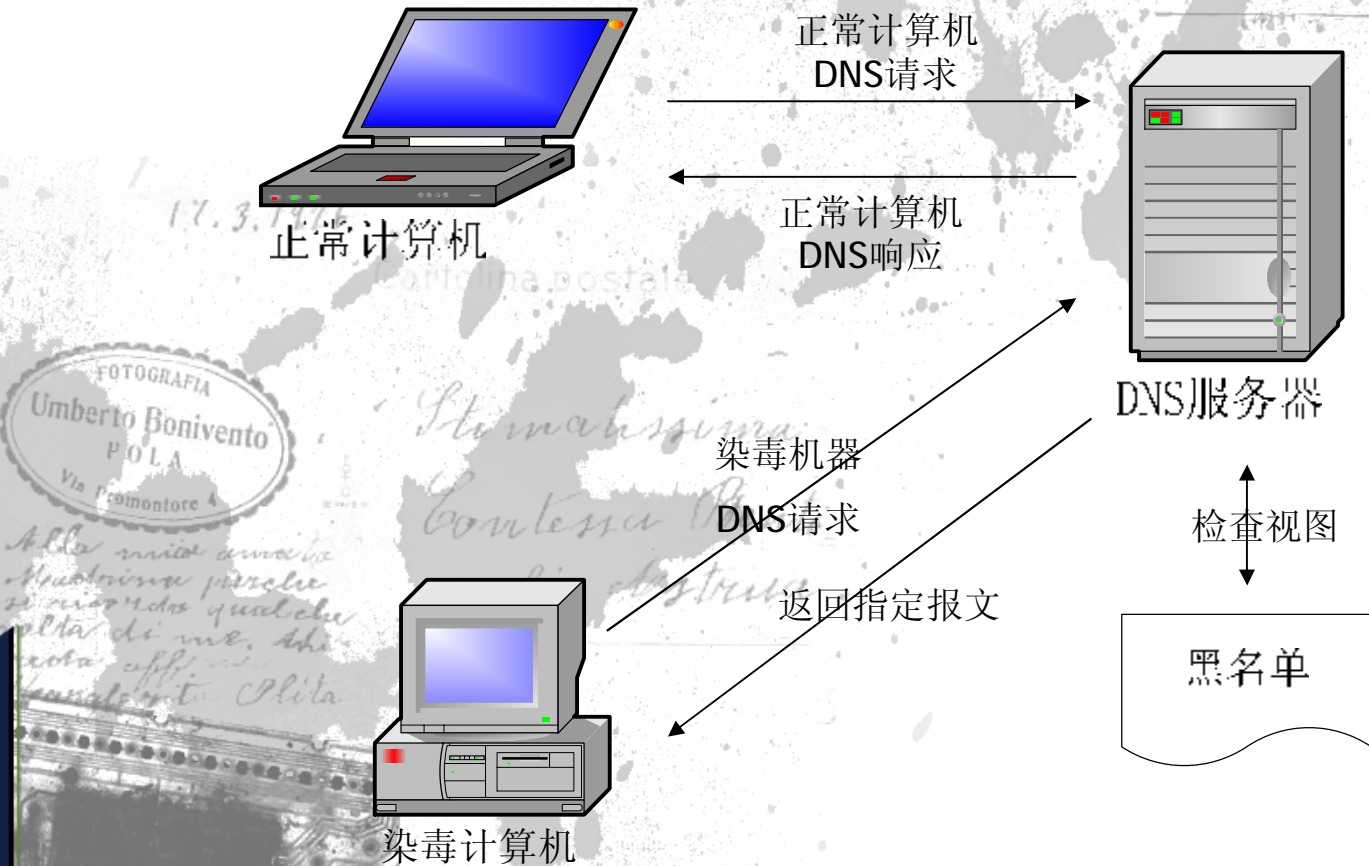
n 工作原理

- DNS服务程序（**BIND9**）支持视图（**view**），对不同的视图，服务程序产生不同的响应；
- 把已感染蠕虫机器的**IP**地址列表放入一个视图中，对这个视图的响应结果设定为指定结果；

n 优缺点

- 仅修改**DNS**配置文件，无需附加程序；
- 操作简单，**DNS**服务管理人员可以自行完成；
- 不同**DNS**服务程序的配置方式不同，很多版本的**DNS**服务程序不支持视图；

配置视图方式流程示意图



端口转发



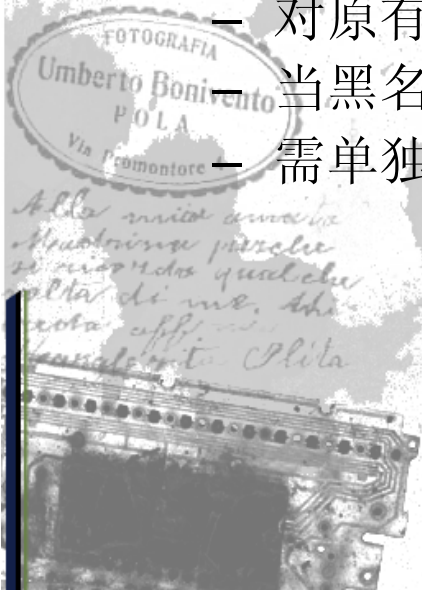
X'CON 2003

n 工作原理

- 端口转发程序代替原有DNS服务监听端口；
- 收到DNS请求时，在指定为文件中查找DNS请求者IP地址；如果在文件中找到DNS请求者IP地址，则返回伪造的DNS响应报文；
- 否则，将请求报文转发给在其他端口（或主机）运行的正常DNS服务程序并将DNS服务程序返回的响应报文转回给DNS请求者。

n 优缺点

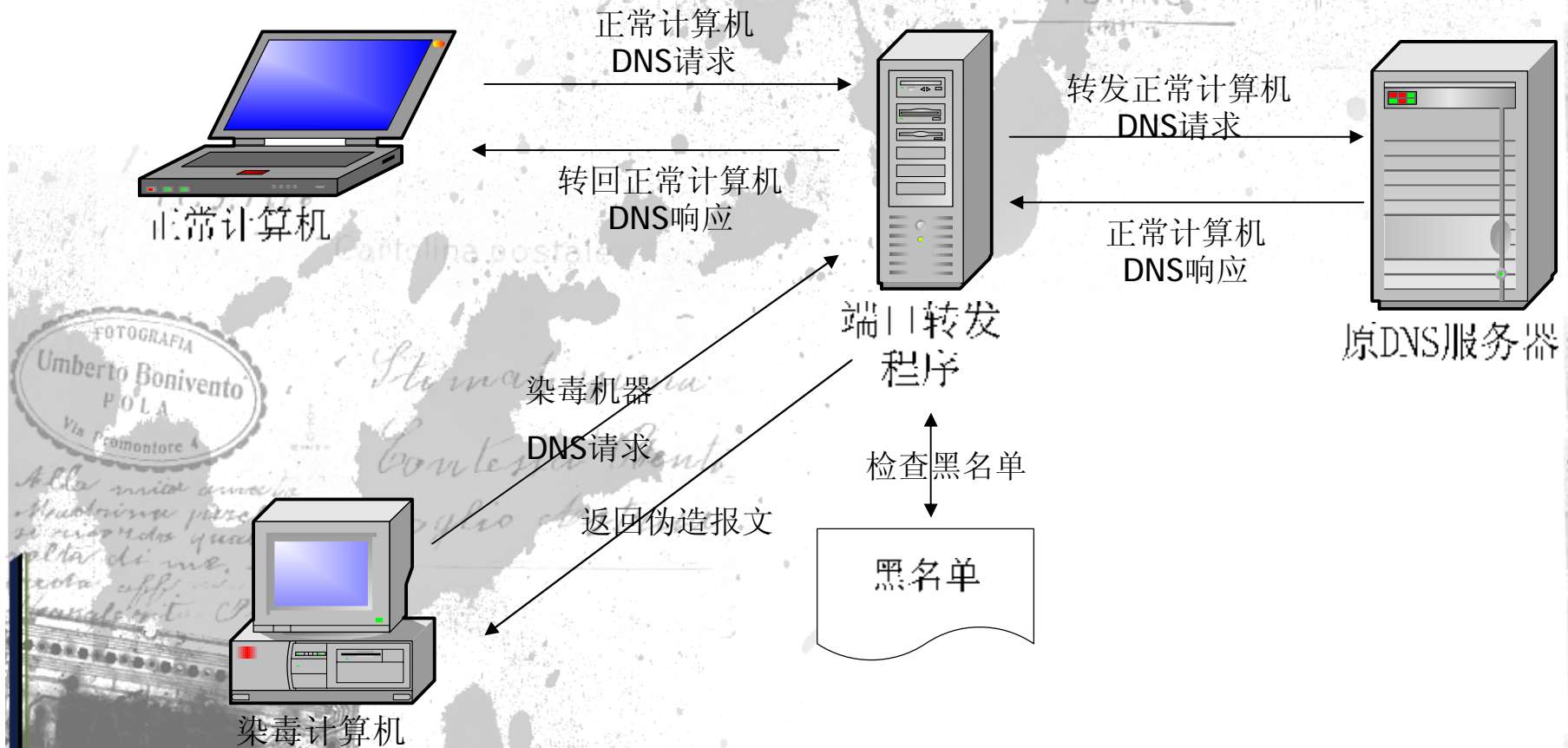
- 对原有系统改动小，可适用于各种不同的DNS服务程序；
- 当黑名单为空时，同原有DNS系统工作效果相同；
- 需单独编程，需要处理和考虑各种复杂情况；





X'CON 2003

端口转发方式流程示意图



系统整体框架设计



X'CON 2003

n 检测服务器（IDS）：

- 定期生成染毒计算机IP地址列表；

n 修改过的DNS服务器：

- 获取染毒计算机IP地址列表
- 过滤染毒计算机IP地址产生的DNS请求；
- 将染毒计算机导向警示服务器；

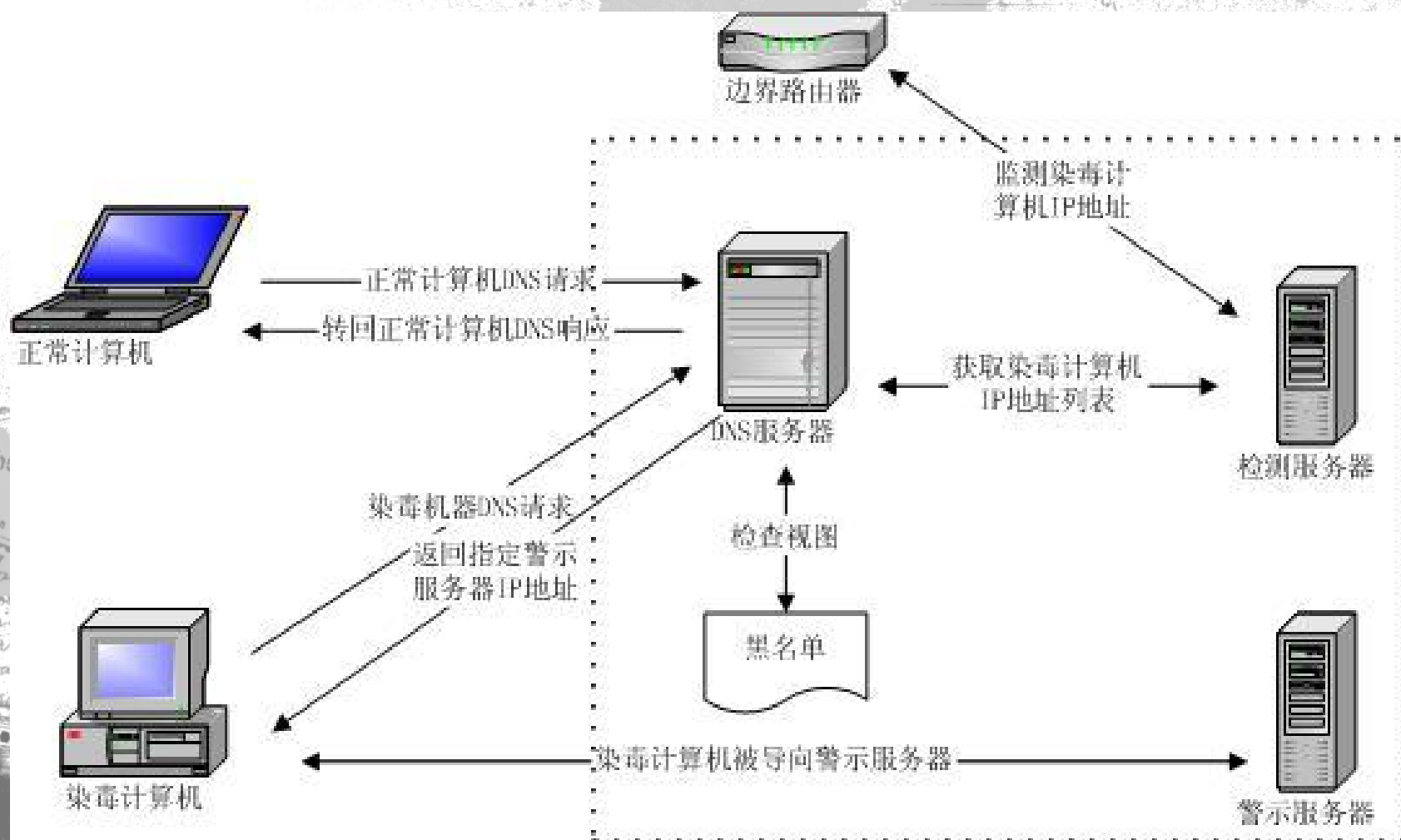
n 警示服务器（Warning Information Server）：

- 提供染毒告警信息；
- 提供补丁程序、杀毒工具下载；
- 收集用户相关信息；



基于配置视图方式的系统结构示意图

X'CON 2003



基于配置视图方式的系统实施方案



X'CON 2003

n 检测服务器

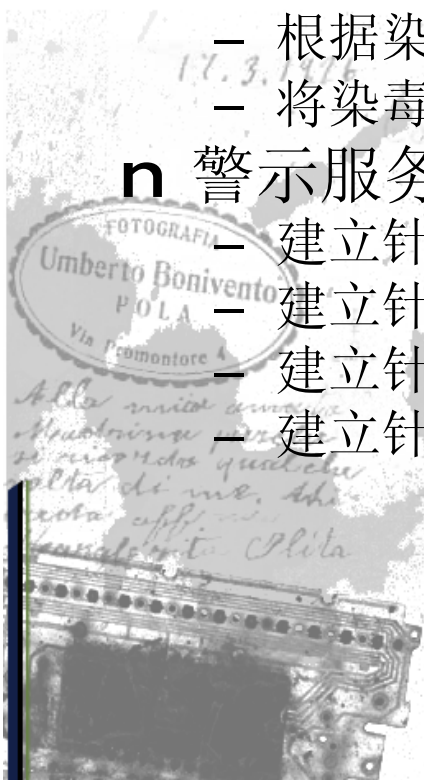
- 配置在边界路由器上，根据蠕虫特征纪录染毒计算机IP地址；

n DNS服务器

- 从检测服务器获取染毒计算机IP地址列表；
- 根据染毒计算机IP地址列表配置视图；
- 将染毒计算机导向警示服务器；

n 警示服务器

- 建立针对HTTP协议的Web警示页面；
- 建立针对Telnet协议的警示信息；
- 建立针对SMTP协议的警示信息；
- 建立针对POP3协议的警示邮件；



检测服务器 (IDS)



X'CON 2003

n 在边界路由器上配置镜像端口:

```
conf
```

```
monitor session 1 source 9/1 destination 9/3
```

n 设置检测程序及检测规则:

```
alert icmp $HOME_NET any -> $EXTERNAL_NET any
```

```
(msg:"Nachi";
```

```
content:"|aaaaaa|";dsize:64;itype:8;offset:1;depth:6;
```

```
reference:arachnids,154; sid:483; classtype:misc-activity;
```

```
rev:2;)
```

n 生成染毒计算机IP地址列表;



BIND 9 视图配置方案 (con.)



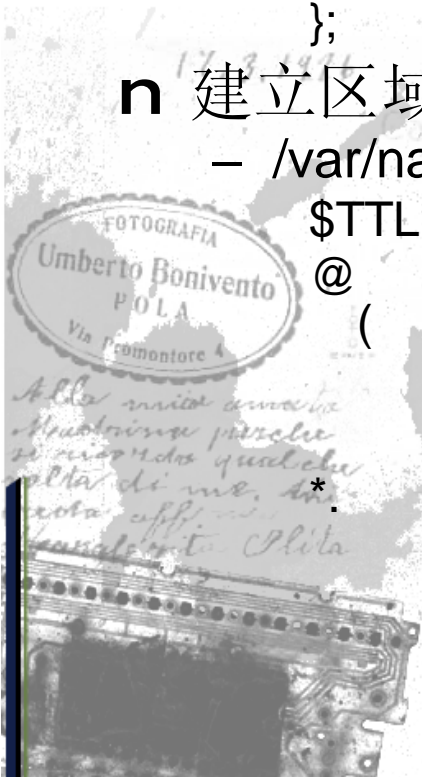
n 建立ACL文件

- 包含染毒计算机IP地址列表;
 - /var/named/ip包含内容如下
- ```
acl "fakesponse" {
 202.112.50.214; # the ip of one infected machine.
};
```

## n 建立区域解析文件

- /var/named/fake.cn包含内容如下

```
$TTL 600
@ IN SOA ccert.edu.cn. hostmaster.ccert.edu.cn
(
 2002031801 28800 1800 604800 86400)
IN NS 127.0.0.1
IN A 202.112.57.9 #the ip of WIS
```



# BIND 9 视图配置方案 (#)



X'CON 2003

## n 修改named.conf

- 引入控制地址列表:

```
include "ip";
```

- 建立如下视图:

```
view "internal" {
 match-clients { "fakesresponse";};
 zone "." in {
 type master;
 file "fake.cn";
 };
```

n 定时更新ACL文件达到动态处理效果



# 警示服务器 (WIS)



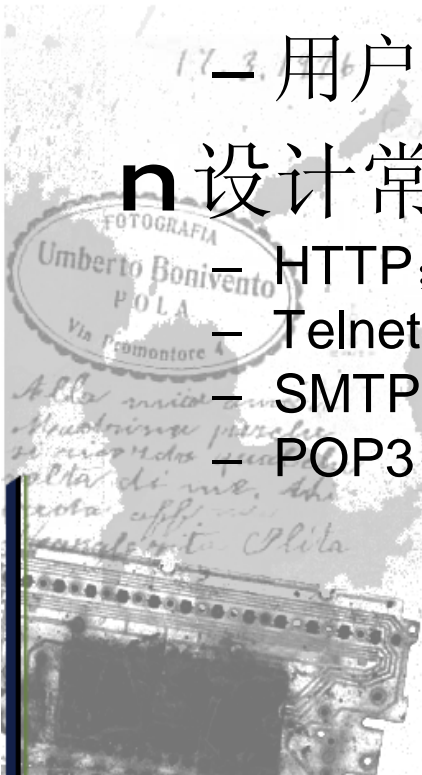
X'CON 2003

## n 制定染毒告警信息:

- 查杀软件下载;
- 补丁软件下载;
- 用户信息收集;

## n 设计常规Internet服务

- HTTP;
- Telnet;
- SMTP;
- POP3;





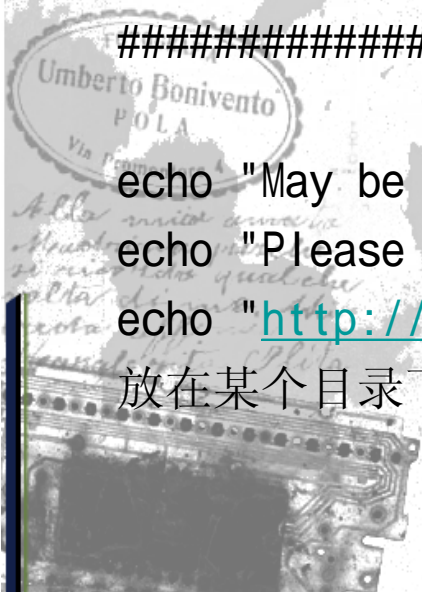
# 警示服务器上针对Telnet协议的警示方案 (con.)



## n编写telnet协议服务程序

```
#!/bin/sh
#####
#Fake Telnetd for warning infected machines! #
17.3.1926
#By Hui ZHENG. <zhenghui_at_cernet.edu.cn> 2003.11.27 #
#####
echo "May be your machine have been infected by Nachi worm!"
echo "Please download pache software from this site!"
echo "http://ccert.tsinghua.edu.cn"
```

放在某个目录下，如/root/DNS/Port23.sh



# 警示服务器上针对Telnet协议的警示方案 (con.)



## n 修改telnetd配置信息

```
[root@spark root]# cd /etc/xinetd.d
[root@spark xinetd.d]# cp telnet telnet.bak
[root@spark xinetd.d]# vim telnet
```

将原有配置:

```
service telnet
{
 disable = yes
 flags = REUSE
 socket_type = stream
 wait = no
 user = root
 server = /usr/sbin/in.telnetd
 log_on_failure += USERID
}
```

修改为:

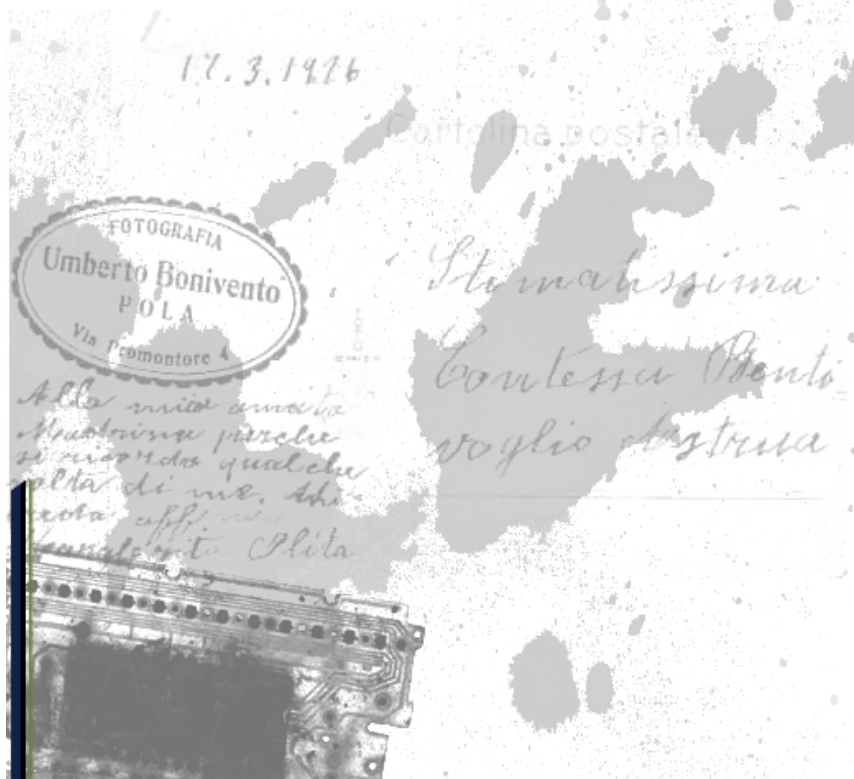
```
service telnet
{
 disable = no
 flags = REUSE
 socket_type = stream
 wait = no
 user = root
 server = /root/DNS/Port23.sh
 log_on_failure += USERID
}
```

# 警示服务器上针对Telnet协议的警示方案 (#)



X'CON 2003

**n** 重新启动xinetd服务  
`/etc/init.d/xinetd restart`



# 警示服务器上针对POP3协议的 警示方案 (con.)



X'CON 2003

n 编写POP3协议服务程序

```
#!/usr/bin/expect
```

```

#Fake POP3d for warning infected machines! #

#By Hui ZHENG. <zhenghui_at_cernet.edu.cn> 2003.12.10 #
#####
```

```
send "+OK Qpopper (version 4.0.5) at ccert.edu.cn starting. <23831.1069924056@ccert.edu.cn>\r\n"
```

```
expect {
"USER" {send "+OK Password required for zhenghui.\r\n";exp_continue}
"PASS" {send "+OK zhenghui has 1 visible message in 575 octets.\r\n";exp_continue}
"STAT" {send "+OK 1 575\r\n";exp_continue}
"UIDL" {send -- "-ERR \r\n";exp_continue}
"TOP" {send -- "-ERR \r\n";exp_continue}
"LIST" {send "+OK 1 visible messages 575 octets\r\n";send "1 372\r\n";send ".\r\n";exp_continue}
"RETR" {send "+OK 575 octets\r\n";send "From: zhenghui@ccert.edu.cn\r\n";
send "Subject: warning\r\n\r\n";
send "May be your computer was infected by Nachi worm!\r\n";
send "Please download patch software from:\r\n";
send "http://www.ccert.edu.cn\r\n";
send ".\r\n";exp_continue}
"DELE" {send "+OK \r\n";exp_continue}
"QUIT" {send "+OK Pop server at ccert.edu.cn signing off.\r\n";close;exp_continue}
}
```

放在某个目录下，如/root/DNS/Port110.sh

# 警示服务器上针对POP3协议的警示方案 (con.)



## n 修改POP3配置信息

```
[root@spark root]# cd /etc/xinetd.d
[root@spark xinetd.d]# cp ipop3 ipop3.bak
[root@spark xinetd.d]# vim ipop3
```

将原有配置:

```
service pop3
```

```
{
 disable = yes
 flags = REUSE
 socket_type = stream
 wait = no
 user = root
 server = /usr/sbin/ipop3d
 log_on_failure += HOST
}
```

修改为:

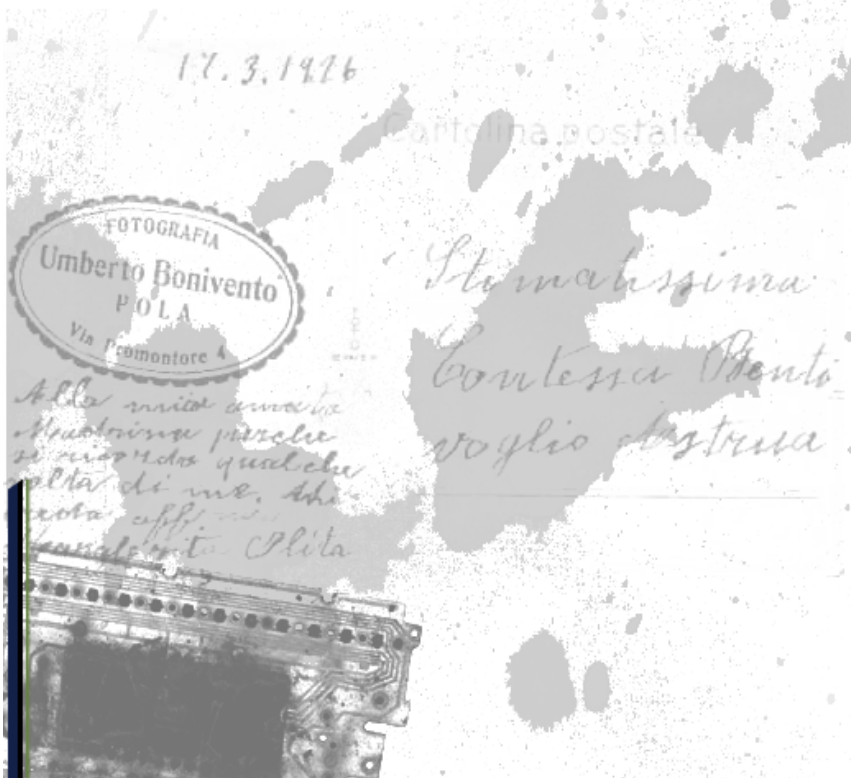
```
service pop3
{
 disable = no
 flags = REUSE
 socket_type = stream
 wait = no
 user = root
 server = /root/DNS/Port110.sh
 log_on_failure += HOST
}
```

# 警示服务器上针对POP3协议的 警示方案 (#)



X'CON 2003

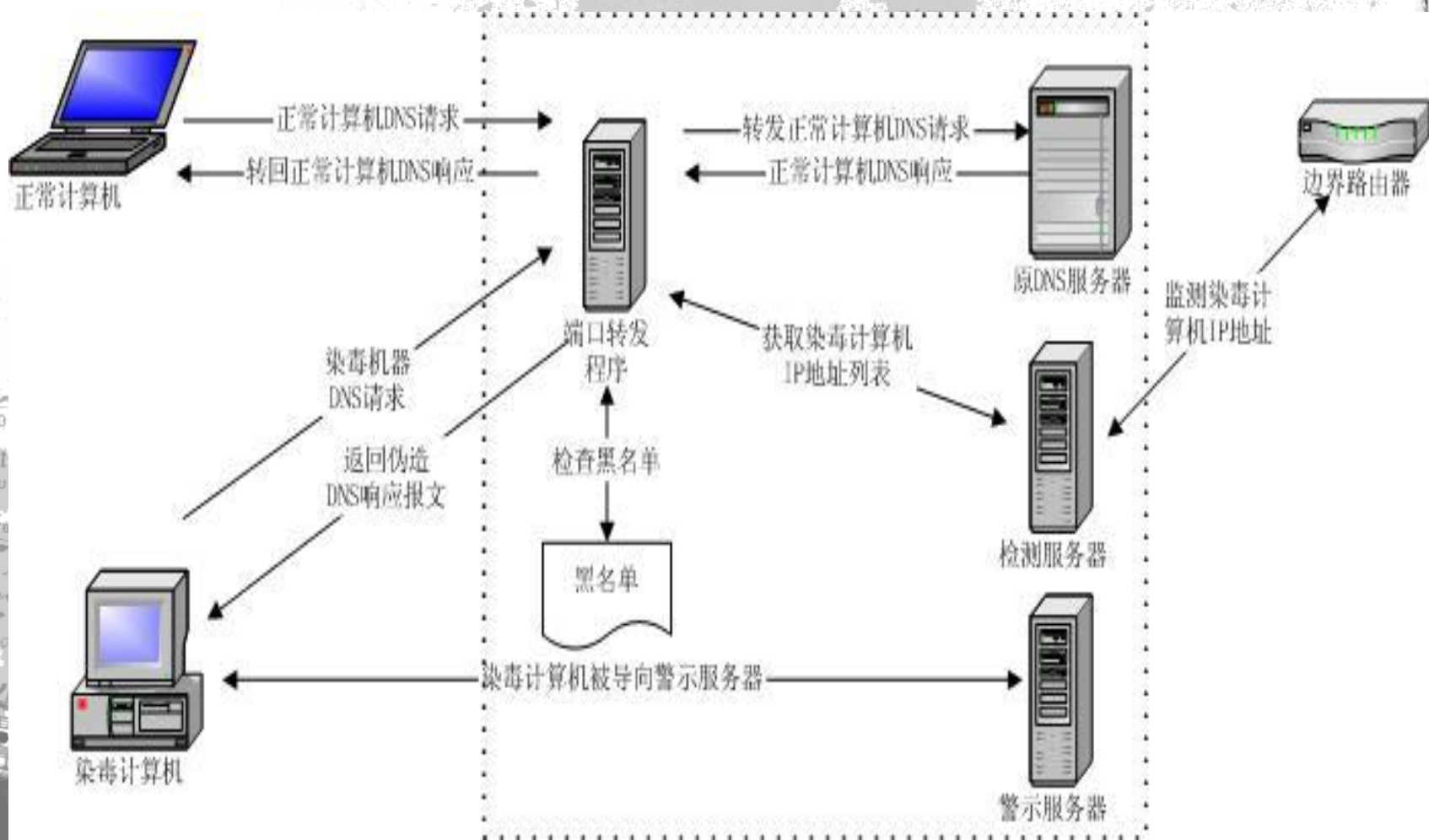
**n** 重新启动xinetd服务  
`/etc/init.d/xinetd restart`



# 基于端口转发方式的系统结构示意图



X'CON 2003



# 基于端口转发方式的系统实施方案



X'CON 2003

## n 检测服务器

- 配置在边界路由器上，根据蠕虫特征纪录染毒计算机IP地址；

## n 端口转发服务器

- 获取染毒计算机IP地址列表；
- 过滤染毒计算机IP地址产生的DNS请求；
- 将正常DNS请求转向DNS服务器；
- 向染毒计算机返回警示服务器IP地址；

## n DNS服务器

- 响应正常DNS请求；

## n 警示服务器

- 建立针对HTTP协议的Web警示页面；
- 建立针对Telnet协议的警示信息；
- 建立针对SMTP协议的警示信息；
- 建立针对POP3协议的警示邮件；



# 配置端口转发服务器 (con.)



X'CON 2003

- n 确定原DNS服务程序的新位置;
  - 在本机另一端口;
  - 在另一IP地址;
- n 确定检测服务器IP地址;
  - 获取染毒计算机IP地址列表;
- n 确定警示服务器IP地址;
  - 作为伪造的DNS请求响应报文返回内容;
- n 在原DNS服务程序端口运行端口转发服务程序;

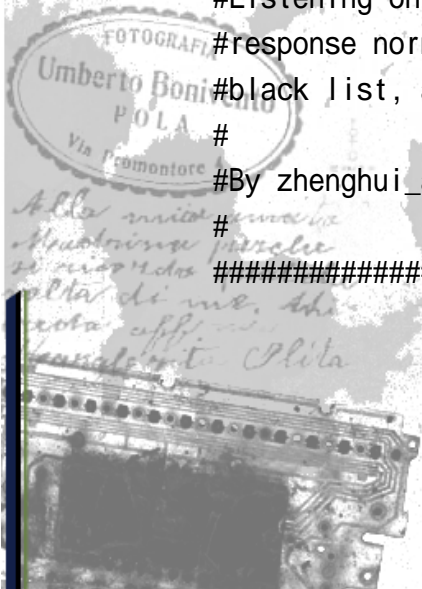
# 配置端口转发服务器 (#)



## nPerl语言实现脚本:

```
#!/usr/bin/perl -w

#####
#DNS isolation concept samples. #
#
#Program Name: Trans.pl #
#
#Function Description: #
#Listening on a port(53), as a DNS server, #
#response normal DNS query. If client in #
#black list, a fake response packet given. #
#
#By zhenghui_at_ccert.edu.cn. 2003.10.29 #
#
#####
```



```

use strict;
use IO::Socket;

my $dns = "202.112.57.6";
my $port = shift || 53;
my $proto = getprotobyname("udp");

my $iaddr = inet_aton($dns);

#wait for client request!.
socket(SERVER,PF_INET,SOCK_DGRAM,$proto) or die "socket: $!";
setsockopt(SERVER,SOL_SOCKET,SO_REUSEADDR,1) or die "setsock: $!";

my $paddr = sockaddr_in($port,INADDR_ANY);

bind(SERVER,$paddr) or die "bind: $!";

#to DNS server.
$paddr = sockaddr_in($port,$iaddr);
socket(DNS,PF_INET,SOCK_DGRAM,$proto) or die "error socket to dns: $!";

my $client_addr;
my $clientbuffer;
my $dnsbuffer;
my $fakeip;
my @unpackdata;
my $i;

while ($client_addr = recv(SERVER,$clientbuffer,4096,0))
{
 $i++;
 my ($client_port,$client_ip) = sockaddr_in($client_addr);
 my $client_ipnum = inet_ntoa($client_ip);
 my $client_host = gethostbyaddr($client_ip,AF_INET);

```

```

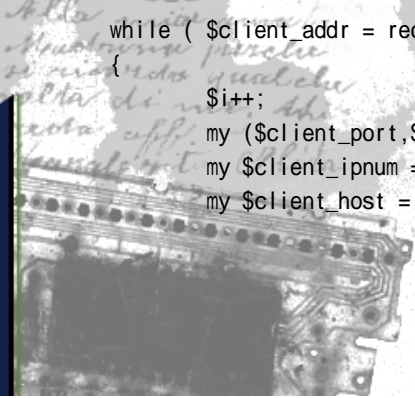
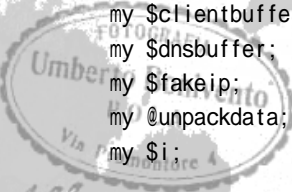
 if (&InFile($client_ipnum))
 {
 @unpackdata = unpack("H4 x11
H*",$clientbuffer);
 #same request!
 $fakeip="8580000100010001000103777770563636572740365647502636e00000
10001c00c00010001000151800004ca703909c0100002000100015180000603646e7
3c010c03e00010001000151800004ca703906";
 $dnsbuffer =
pack("H4H*",$unpackdata[0],$fakeip);
 send(SERVER,$dnsbuffer,0,$client_addr);
 next;
 }
 print "got a connection from :
$client_host",[$client_ipnum]\n";
 send(DNS,$clientbuffer,0,$paddr) || warn " send to
dns error: $! \n";
 recv(DNS,$dnsbuffer,4096,0);
 send(SERVER,$dnsbuffer,0,$client_addr)|| warn " send
to client error: $! \n";
}

sub InFile
{
 my $ip = shift;
 open(IN,"blacklist.txt");

 while(<IN>)
 {
 if (/^ip/)
 {
 close(IN);
 return 1 ;
 }
 }

 close(IN);
 return 0;
}

```

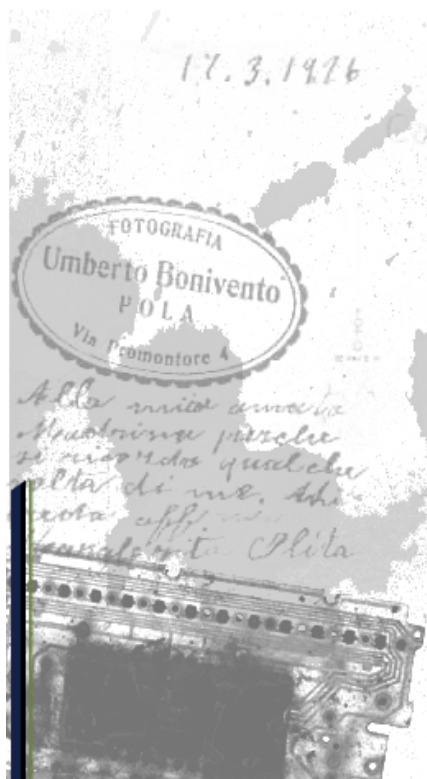


# 基于配置视图方式的系统性能分析



X'CON 2003

**n** 同原DNS系统相同;

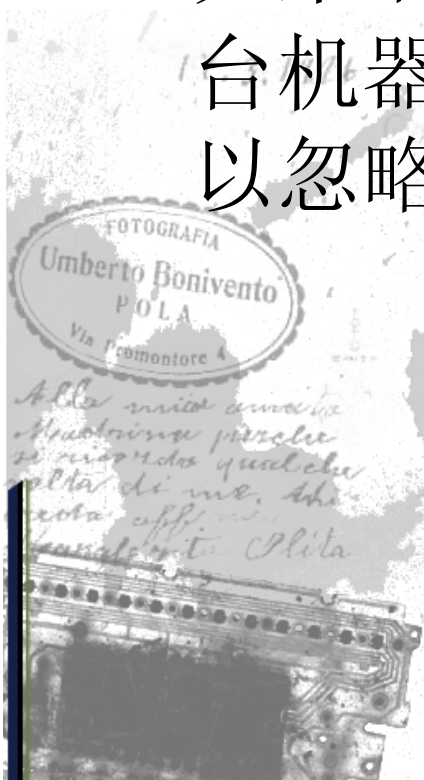


# 基于端口转发方式的系统性能分析



X'CON 2003

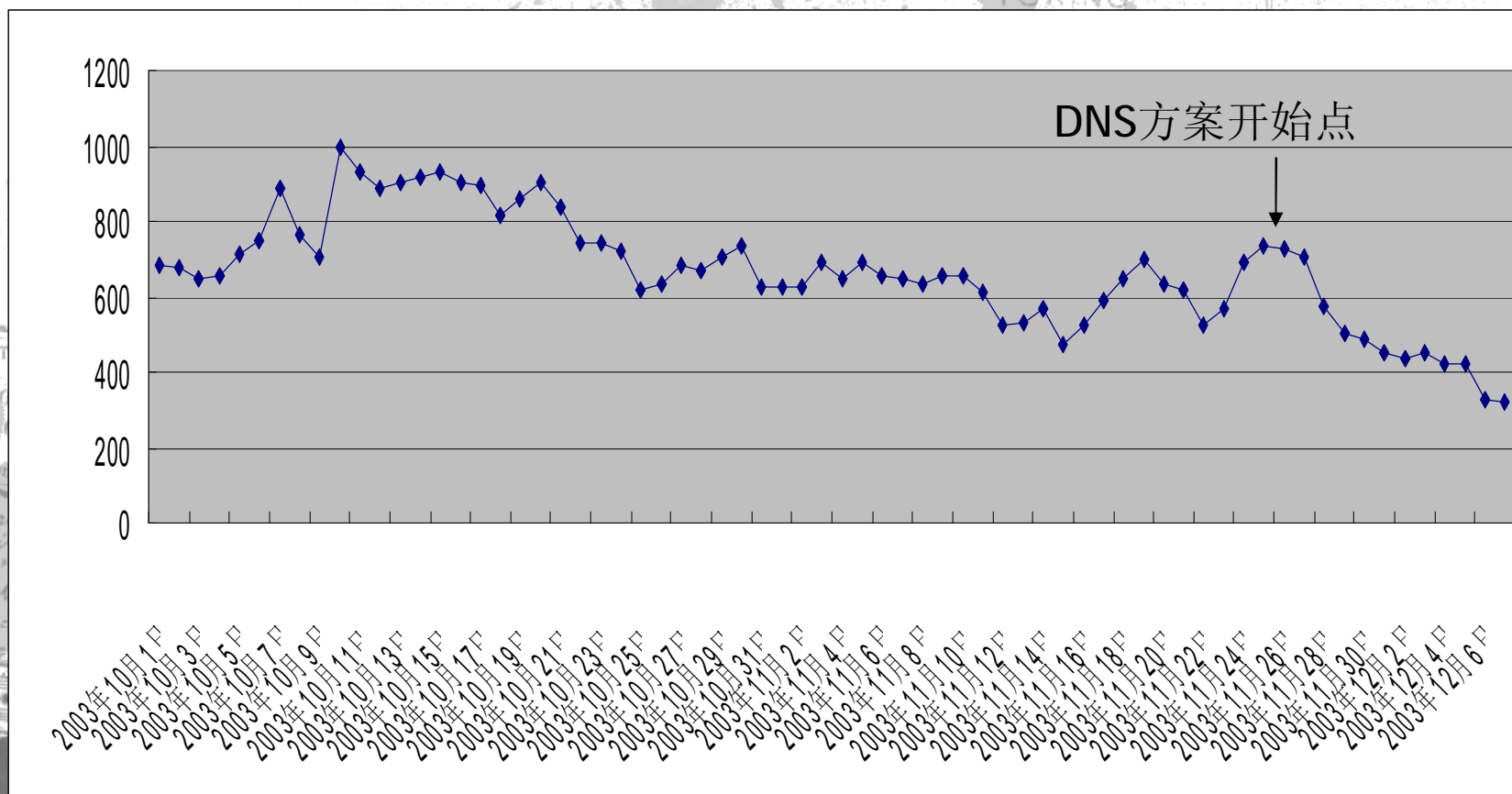
- n** 延迟取决于端口转发服务器同原DNS服务器之间网络通信时间；
- n** 如果端口转发程序同原DNS服务器在同一台机器上，则端口转发程序造成的时延可以忽略不计；



# 清华校园网实施效果 (1)



n 总体效果:



# 清华校园网实施效果 (2)



## n DNS方案实施力度衡量公式:

- 令集合 $I(k)$  表示每天检测到的染毒计算机IP地址;
- 令集合 $B(k)$  表示每天访问警示服务器(WIS)的计算机IP地址;
- 则DNS方案的实施力度可由 $S=B(k)/I(k)$  来衡量。

## n 影响实施力度的因素:

- 网络中有多台DNS服务器, 只在其中某几台实施;
- 实施时间不连续;
- 其他控制措施的影响, 例如ACL使DNS无法正常获得检测地址列表;

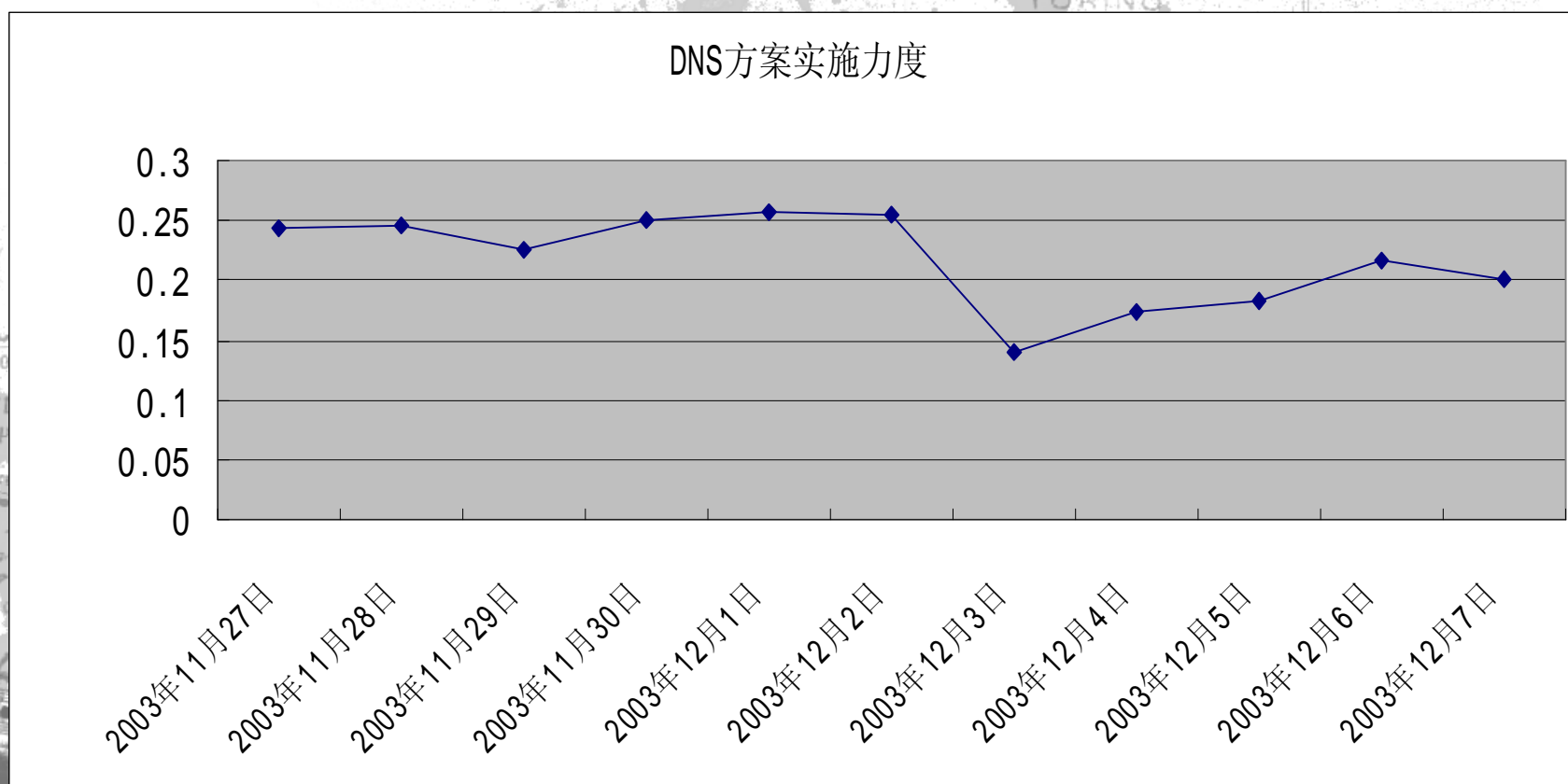


# 清华校园网实施效果 (3)



X'CON 2003

n DNS方案实施力度: (较低)





# 清华校园网实施效果 (4)



## n DNS方案效果衡量公式:

- 被导向到WIS的计算机如果在将来又出现在每天检测到的染毒计算机IP地址列表中, 则表明DNS方案对此计算机无效, 无效计算机数目与被导向到WIS的计算机数目的比率可以用来衡量DNS方案的有效性。

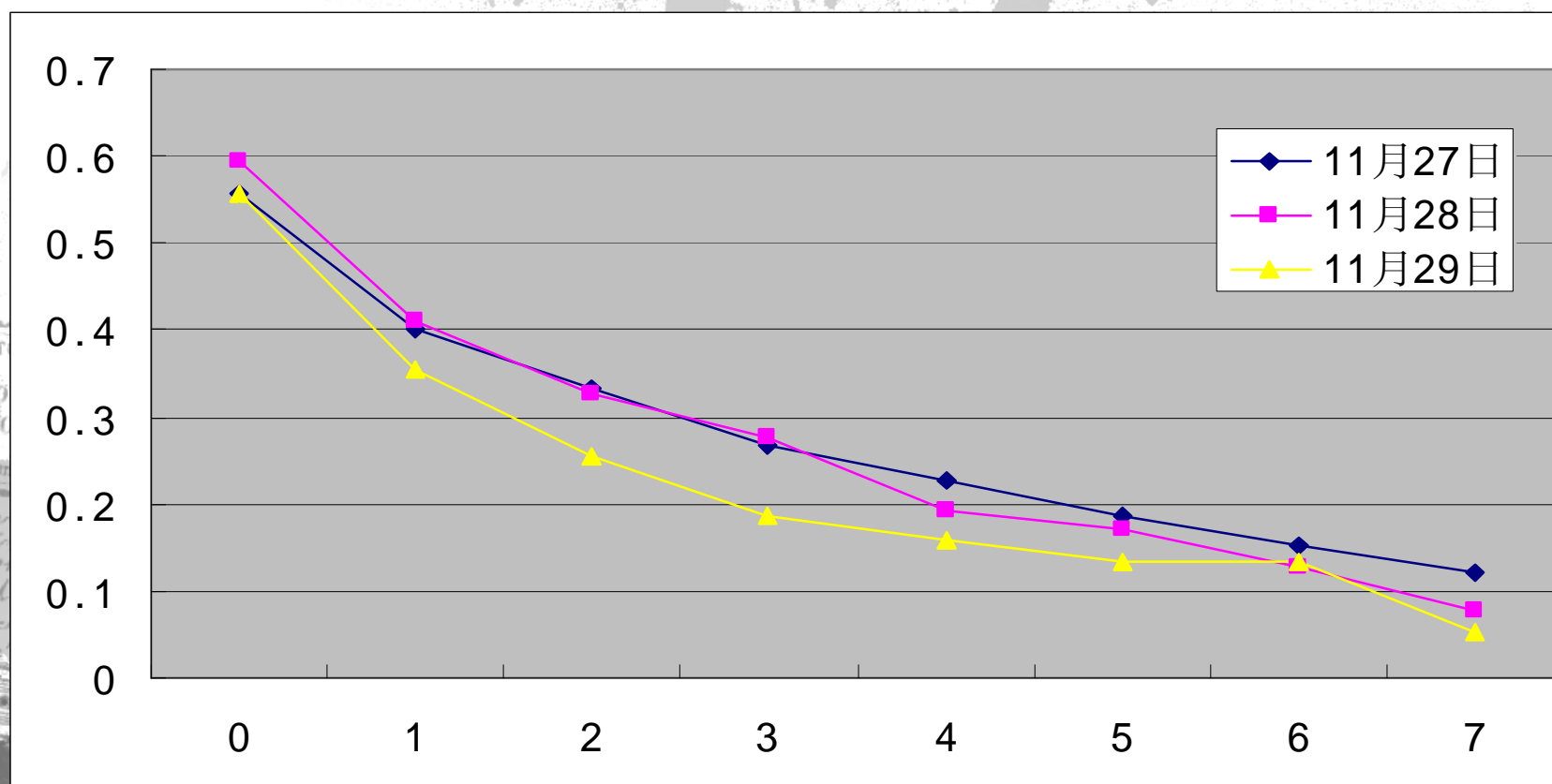
$$E(k) = 1 - \frac{\sum_{t=k+1}^N \|B(k) \cap Y I(t)\|}{\|B(k)\|}$$

# 清华校园网实施效果 (5)



X'CON 2003

n DNS方案效果：3~4天，80%被清除。



# 参考文献



X'CON 2003

## n 原始数据:

- 段海新, 清华校园网Nachi蠕虫监测系统;
- 郑先伟, 清华校园网应急响应网站访问日志;

n 孙彬, “DNS配置方法”, CCERT内部资料。

n 雷迎春, 龚奕利 译。Paul Albitz & Cricket Liu 著。《DNS与BIND》。中国电力出版社, 2002。

n RFC1939 (POP3),

<http://www.fanqiang.com/a6/b9/20010929/1305001372.html>

POST CARD

Carte postale

Carte postale - Correspondence



X'CON 2003

STUDIO FOTOGRAFICO  
DANTE MORONI  
Via S. ... N. 36  
TORINO

Thanks !

17.3.1926

Cartolina postale

FOTOGRAFIA  
Umberto Bonivento  
PIOLA  
Via Piemontese 4

Stimabatissima  
Contessa Bentivoglio  
voglio abbracciarti.

Alle mie amiche  
Maurina e Paola  
se ricordo qualche  
volta di me. Ah  
non affrettarsi  
lasciatevi Plita

