

Setiri: Advances in Trojan Technology

Roelof Temmingh
Haroon Meer
BlackHat Asia 2002



Schedule

Introduction

Why Trojans?

Brief History of Trojans & Covert Channels

The Hybrid model

Setiri: Advances in Trojan Technology

Demonstration

Taking it further

Possible fixes



Introduction

SensePost

The speakers

Objective of presentation



Why Trojans?

Profile of Trojan users

Real criminals...

...don't write buffer overflows

The weirdness of the industry

Examples



USA • EUROPE • ASIA
BlackHat

digital self defense



Brief History of Trojans & Covert Tunnels

Trojans

From Quick Thinking Greeks ...
to Quick Thinking Geeks

Tunnels

Covert Channels



Trojans..

Valid IP – No Filters

Valid IP – Stateless Filters

Private Addresses – Stateful Filters

Private
+ Stateful

+ IDS + Personal Firewalls

+ Content Checking

+ ...



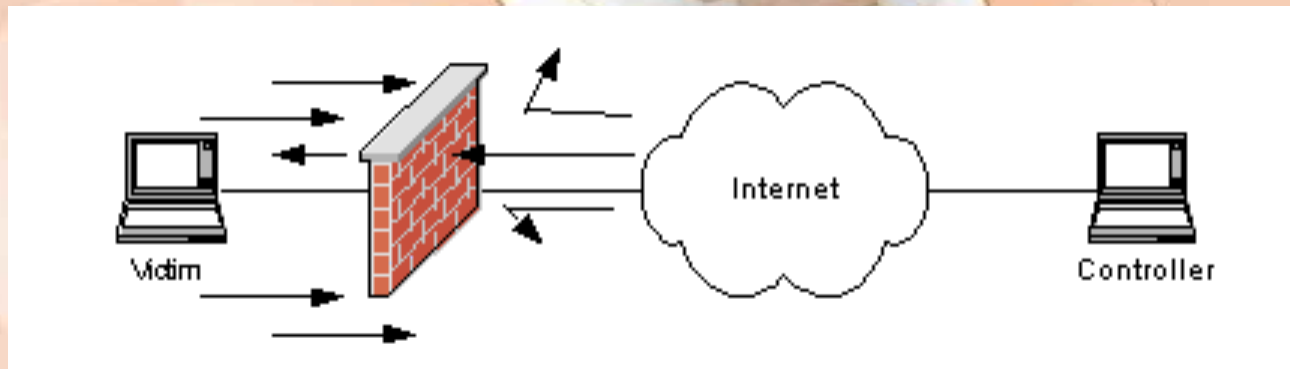
Trojans.. (Valid IP – No Filters)



“get real..”

Trojans..

(Valid IP – Stateless Filter)

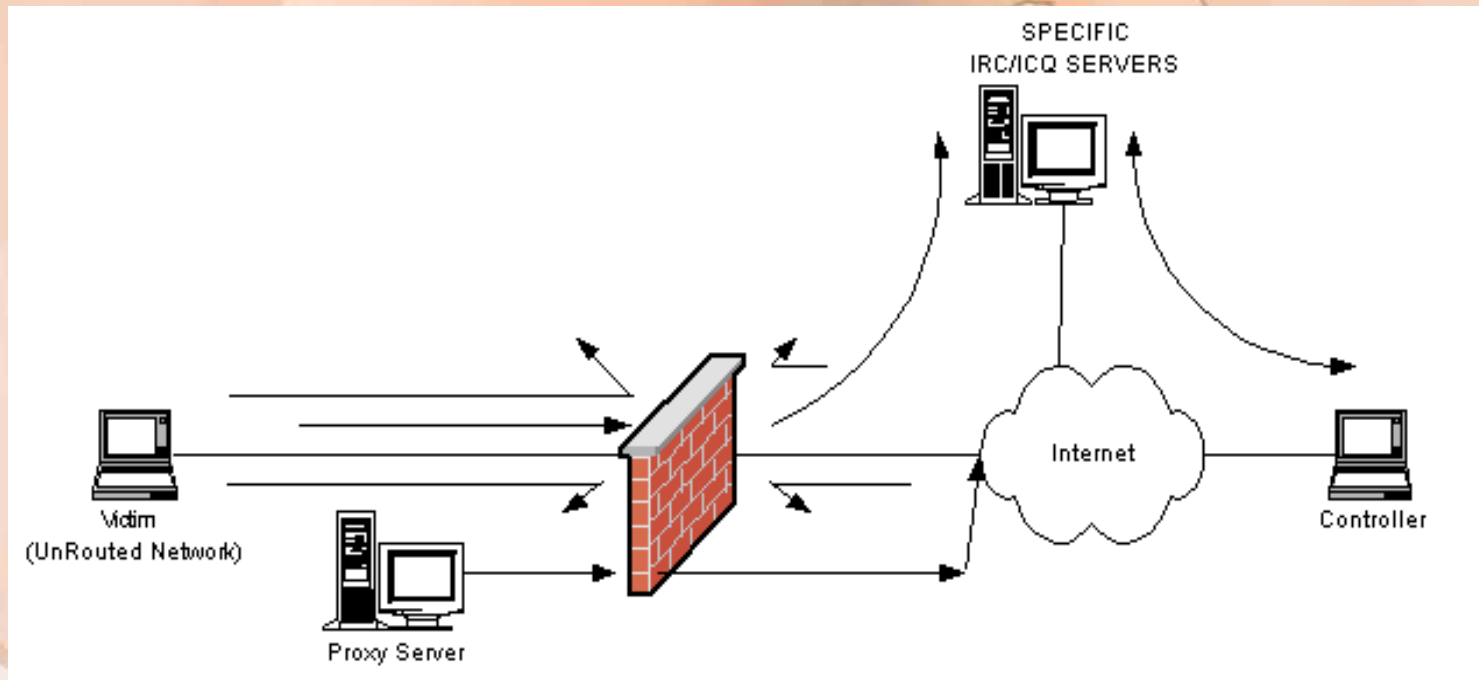


Dial Home Trojans

Random Ports / Open Ports / High Ports
[cDc]

ACK Tunneling
[Arne Vidstrom]

Trojans.. (Stateful Filters)



Back Orifice - <http://bo2k.sourceforge.net>

Gbot
Rattler



Brief History of Trojans & Covert Tunnels

Trojans

From Quick Thinking Greeks ...
to Quick Thinking Geeks

Tunnels

Covert Channels



Tunnels & Covert Channels

1985 – TSC Definition “Covert Channels”

1996 – Phrack Magazine – *LOKI*

1998 – RWWWShell – THC

1999 - HTTPTUNNEL – GNU

2000 - FireThru - Firethru



Conventional Trojans & how they fail

Stateful firewall & IDS

Direct model

Direct model with network tricks

ICMP tunneling

ACK tunneling

Properly configured stateful firewall

IRC agents +

Authentication proxy

HTTP tunnel ++

Personal firewall & Advanced Proxy

HTTP tunnel with Authentication +++



Hybrid model: “*GatSlag*”

Combination between covert
Tunnel and Trojan

Defenses mechanisms today:

Packet filters (stateful) / NAT

Authentication Proxies

Intrusion detection systems

Personal firewalls

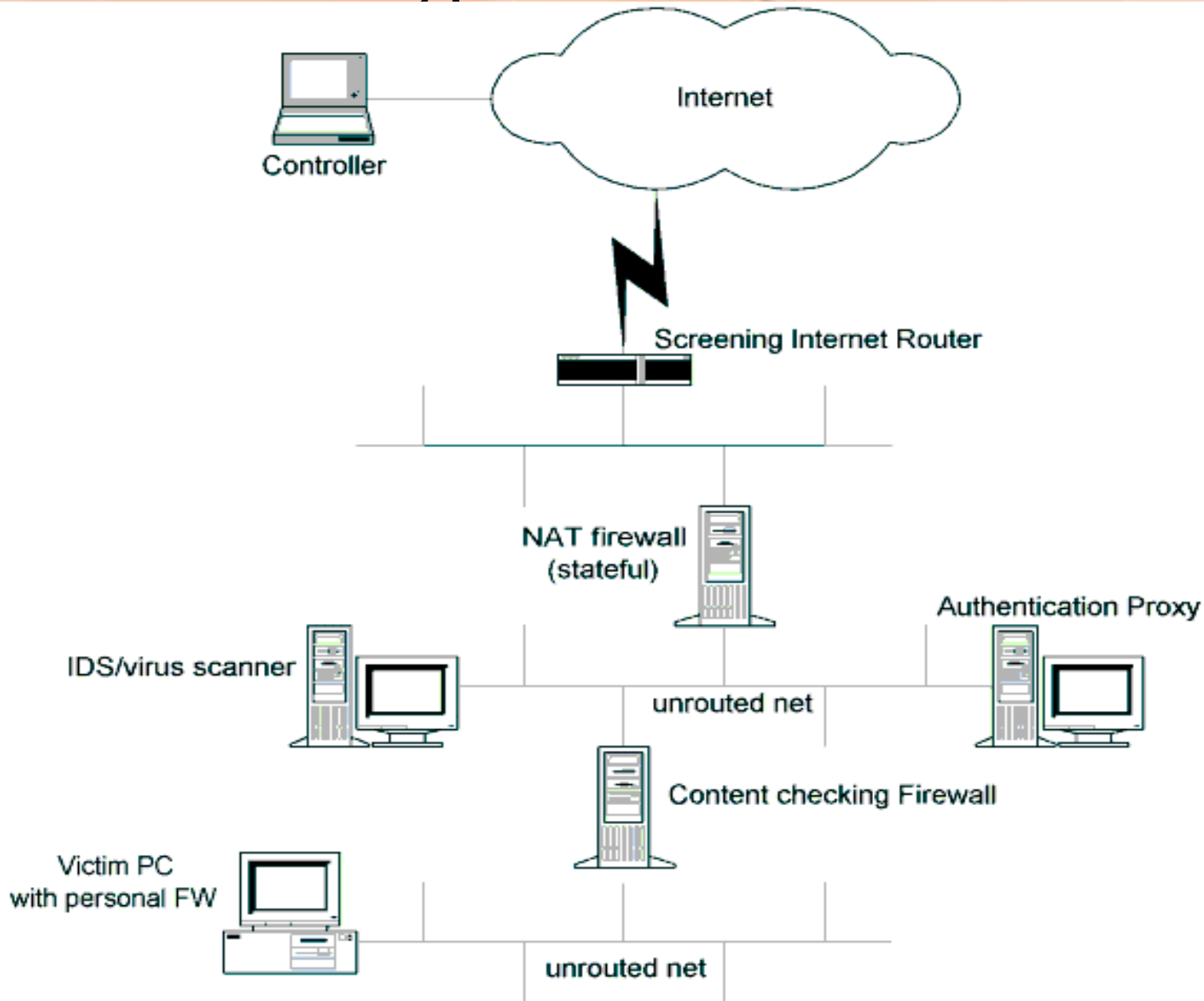
Content/protocol checking

Biometrics/Token Pads/One time passwords

Encryption



A typical network



How *GatSlag* worked

Reverse connection
HTTP covert tunnel

Microsoft Internet Explorer as transport

Controls IE via OLE

Encapsulate in IE, not HTTP

Receive commands in title of web page

Receive encoded data as plain text in body of web page

Send data with POST request

Send alive signals with GET request



Why *GatSlag* worked

Integration of client with MS Proxy

NTLM authentication

SSL capable

Registry changes

Personal firewalls

Just another browser

Platform independent

IE on every desktop

Specify Controller

Via public web page – the MASTER site



How *GatSlag* worked II

Creates invisible browser

Find controller at MASTER

Send request to Controller

If no Controller && retry>7, go to MASTER

Receive reply

Parse reply:

+ Upload file()

+Download file

+Execute command

Loop



Why defenses fail

Firewalls (stateful/NAT)

Configured to allow user or proxy out

Content level & IDS

Looks like valid HTTP requests & replies

Files downloaded as text in web pages

No data or ports to lock on to

SSL provides encryption

Personal firewalls

IE valid application

Configured to allow browsing

Authentication proxies

User surf the web



Problems with *Gatslag*

The Controller's IP can be obtained !

Handling of multiple instances

GUI support

Controller needed to be online

Batch commands

Command history

Multiple controllers

Upload facility not efficient

Platform support

Stability

Session level tunneling



Setiri: Advances in Trojan Technology

Design notes:

Web site contains instructions
CGIs to create new instruction

Controller's interface:

- EXEC (DOS commands)
- TX (File upload)
- RX (File download)

Directory structure – each instance

Trojan “surfs” to web site – just a normal user would



Setiri: Advances in Trojan Technology II

Anonymity

Problems with normal proxies

Already using a proxy

Proxy logs

“Cleaners” provide anonymity

“In browser proxy” – Anonymizer

Trojan -> Cleaner: SSL

Cleaner -> Controller: SSL

Challenges:

Browser history

Temporary files




Setiri Command - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print Mail

Address <http://hackrack.com/cgi-bin/setiri/mainpane.pl?instance=> Go Links >>

SETIRI


[XXXTEST]

[Check pending](#)

[06.21;11.11.32 \[exe\] set](#)
[06.24;11.37.54 \[exe\] dir c:\](#)
[06.24;11.38.12 \[exe\] dir c:*.doc /s](#)
[06.24;11.38.49 \[rx\] testtrf.rtf.txt \(c:\temp\testtrf.rtf\)](#)
[06.24;11.39.22 \[bd\] c:\dir.txt \(pending\)](#)

[Manage another Instance](#)

File upload

exe Parameters

Directory of c:\gatslag

01/04/2002	07:02p	264,192	BH2002.doc
02/03/2002	07:43a	51,200	How GS works.doc
01/04/2002	03:28p	163,840	paper.doc
3 File(s)		479,232	bytes

Directory of c:\MSSQL7\DevTools\Samples\backup




10/31/1998	12:39a	641,536	vbackup.doc
1 File(s)		641,536	bytes

Directory of c:\Perl561\eg\PDK\Re

02/01/2002	03:53p	40,960	TestRe.doc
1 File(s)		40,960	bytes

Directory of c:\Program Files\ActiveState Perl Debugger

Index of /XXXTEST

Name	Last modified	Size	Description
 Parent Directory	19-Jun-2002 08:39	-	
 commands/	20-Jun-2002 20:27	-	
 download/	18-Jun-2002 09:04	-	



Done Internet

Setiri Command - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media

Address <http://hackrack.com/cgi-bin/setiri/mainpane.pl?instance=> Go Links

[XXXTEST] [Check pending](#)

06.24;12.12.18 [exe] dir c:/*.*
 06.24;12.16.01 [b]
 c:\COMPAREXPAYMENTINSTRUCTIONFORM.doc

[Manage another Instance](#)


File upload

Parameters

Downloaded file successfully - check upload directory.

CORPORATE REGULATIONS
Retirement Fund Withdrawal Form

Restricted
For Internal Use Only



COMPAREX RETIREMENT FUND: WITHDRAWAL FORM AND PAYMENT INSTRUCTIONS

PERSONAL DETAILS:

Member's Name:		Employee Number:	
Tax Office:		Tax Ref. Number:	
Company / BU:		Division/Department:	

BANKING DETAILS:

Name of Bank:		Name of Branch:	
Branch Code:		Account Number:	
Name of Account Holder:			
Account Type:	Savings	Current	Transmission

CONTACT DETAILS:

Telephone Number:		Cell. Number:		E-Mail:	
Residential Address:					
Postal Address:					

AUTHORISATION:

Done Internet

#Randomdata# #[Protected by-proxy30.anonymizer.com] - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print Mail News RSS Feeds

Address <https://ssl.anonymizer.com/https://hackrack.com/TEST/commands/commands.asp?KODCDJIXMKEMZS> Go Links

```

?
!iojfvrvwn!#reset#clear#06.20;09.13.27##
!szdktqgsbn!#exe#set#06.21;11.11.11##
!gruzxilyny!#exe#dir c:\#06.24;11.30.11##
!fhakqdmupv!#exe#dir c:\*.*\s#06.24;11.30.11##
!hsfnytinwn!#exe#testrtf.rtf.txt#06.24;11.30.11##
!hlpzyehqt!#exe#dir.txt#06.24;11.30.11##
  
```

Done

<capture> - Ethereal

File Edit Capture Display Tools Help

No.	Time	Source	Destination	Protocol	Info
636	118.091316	draadloos.sensepost.c	168.143.113.14	TCP	3898 > https [SYN] Seq=
638	118.801161	draadloos.sensepost.c	168.143.113.14	TCP	3898 > https [ACK] Seq=
639	118.917482	draadloos.sensepost.c	168.143.113.14	TCP	3898 > https [PSH, ACK]
642	119.811248	draadloos.sensepost.c	168.143.113.14	TCP	3898 > https [PSH, ACK]
647	120.547352	draadloos.sensepost.c	168.143.113.14	TCP	3898 > https [PSH, ACK]
649	121.336857	draadloos.sensepost.c	168.143.113.14	TCP	3898 > https [PSH, ACK]
656	122.600244	draadloos.sensepost.c	168.143.113.14	TCP	3898 > https [ACK] Seq=

Flags: 0x04
 Fragment offset: 0
 Time to live: 128
 Protocol: TCP (0x06)
 Header checksum: 0x93ba (correct)
 Source: draadloos.sensepost.com (196.30.67.67)
 Destination: 168.143.113.14 (168.143.113.14)

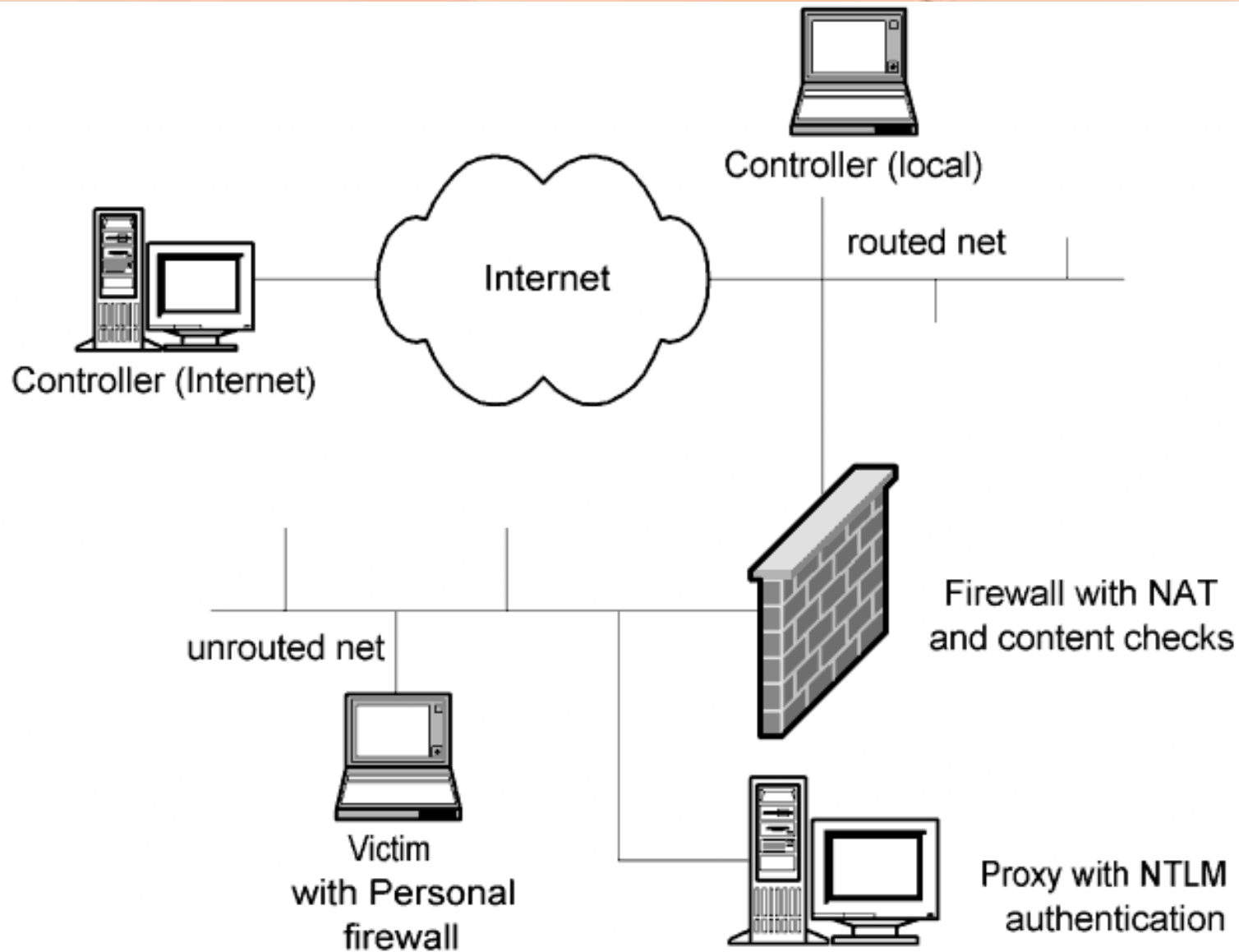
Transmission Control Protocol, Src Port: 3898 (3898), Dst Port: https (443), Seq: 2415450251
 Source port: 3898 (3898)
 Destination port: https (443)
 Sequence number: 2415450251
 Acknowledgement number: 1080955183
 Header length: 20 bytes

```

0000  00 60 97 2a 76 2a 00 d0 59 0f e2 f8 08 00 45 00  . . *v*.. Y....E.
0010  00 28 46 16 40 00 80 06 93 ba c4 1e 43 43 a8 8f  .(F.@... ..CC..
0020  71 0e 0f 3a 01 bb 8f f8 d8 8b 40 6e 11 2f 50 10  g..... ..@n./P.
0030  44 70 7f 4e 00 00                                     Dp0N..
  
```

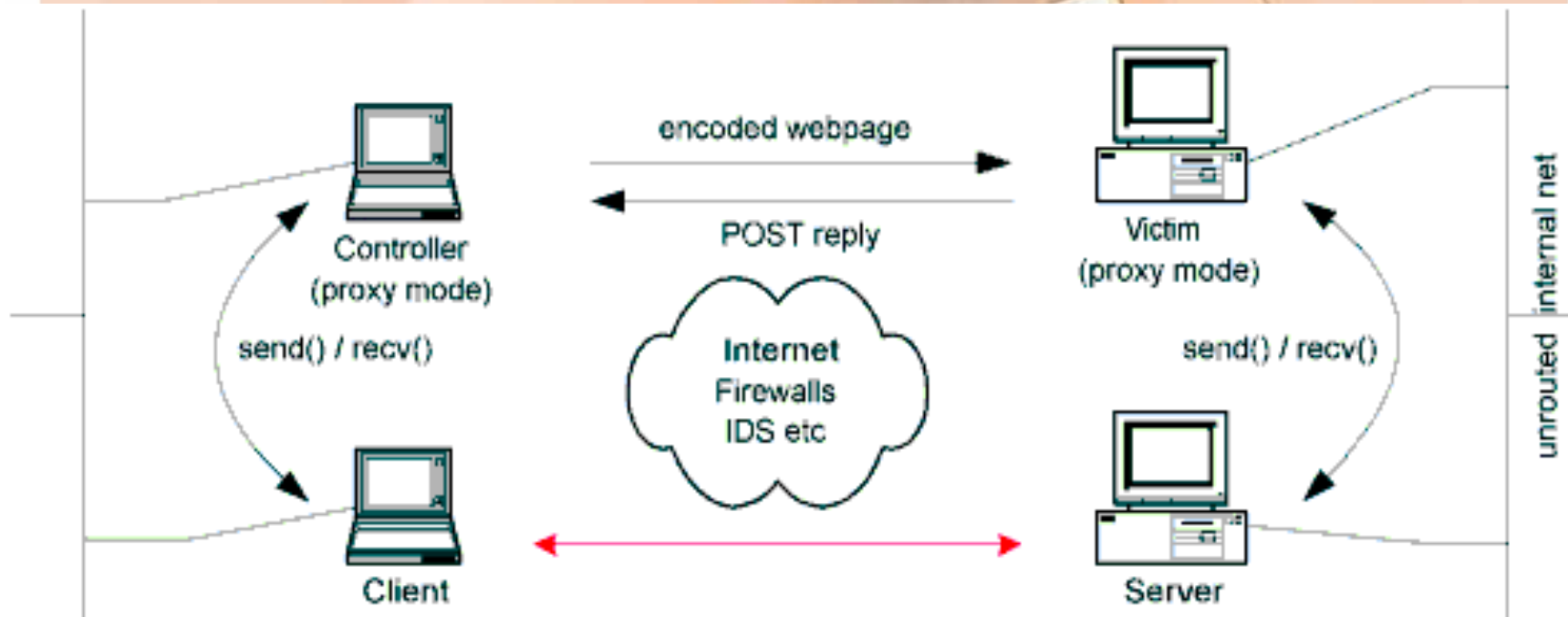
Filter: tcp.dstport == 443 Reset Destination (ip.dst)

Demonstration



Taking it further

Session level tunneling



Flow control challenges

How this is different from HTTP tunneling

A browser is not a socket

No select on browser

Train model

The Controller side

Cannot “send”

Buffering of data at Controller

The Trojan side

Multi-part POSTs

Multiple connections (HTTP)

True network level tunneling



Solving the dilemma

Delivery

White listing

User education

AV, personal firewalls

Should you allow everyone to surf the 'net?



Conclusion

Awareness
Our motivation

