

Network design for ineffective HTTP traffic filtering

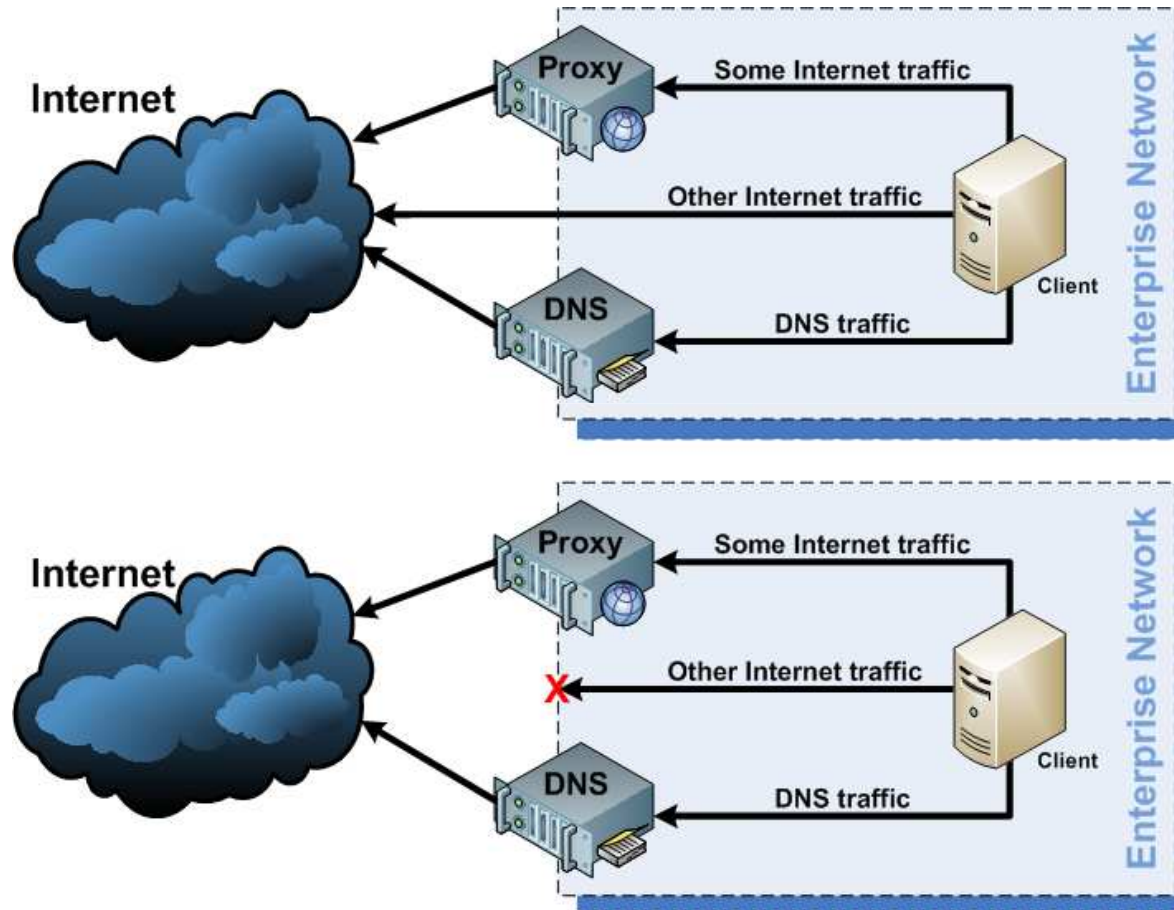
Or, the myriad ways users and apps are bypassing your web filtering proxy

Jeff.Forrstal@Zscaler.com

Quick truths

- Companies want URL filtering, Web 2.0/RIA management, malware mitigation (HTTP A/V), DLP, bandwidth control, P2P containment, etc.
- Typical HTTP-centric solutions are deployed as an HTTP proxy (transparent or direct) on a network that normally allows users to directly access the Internet
- Network protocol/traffic/port whitelisting is seen as scary to many organizations, so they prefer to fail-open rather than fail-closed

Open vs. closed network



Why/how apps are egressing

- The app ignores proxy settings/is not proxy aware
- The user didn't separately configure the app to use a proxy
- The app tunnels through HTTPS-capable proxy ("CONNECT tunnel")
- The app's traffic is wrapped in SSL/TLS
- The app is using UDP

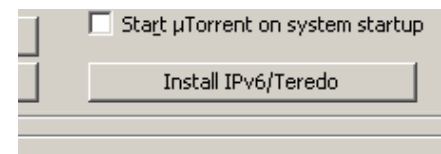
In open networks, the apps "just work" out of the box

Some case studies



Skype uses configured HTTP proxy if direct UDP and TCP connections fail (if direct connection works, might as well use it)

P2P clients now ship with built-in Teredo (IPv6 over UDP) capabilities, for all your tunneling needs



Blocking BitTorrent HTTP tracker access is effective, but negated if client can use DHT (via UDP) to find peers

Visualization of the typical security effectiveness / comprehensiveness of a filtering HTTP proxy deployed on an otherwise open network:



What about...

Direct browser configuration

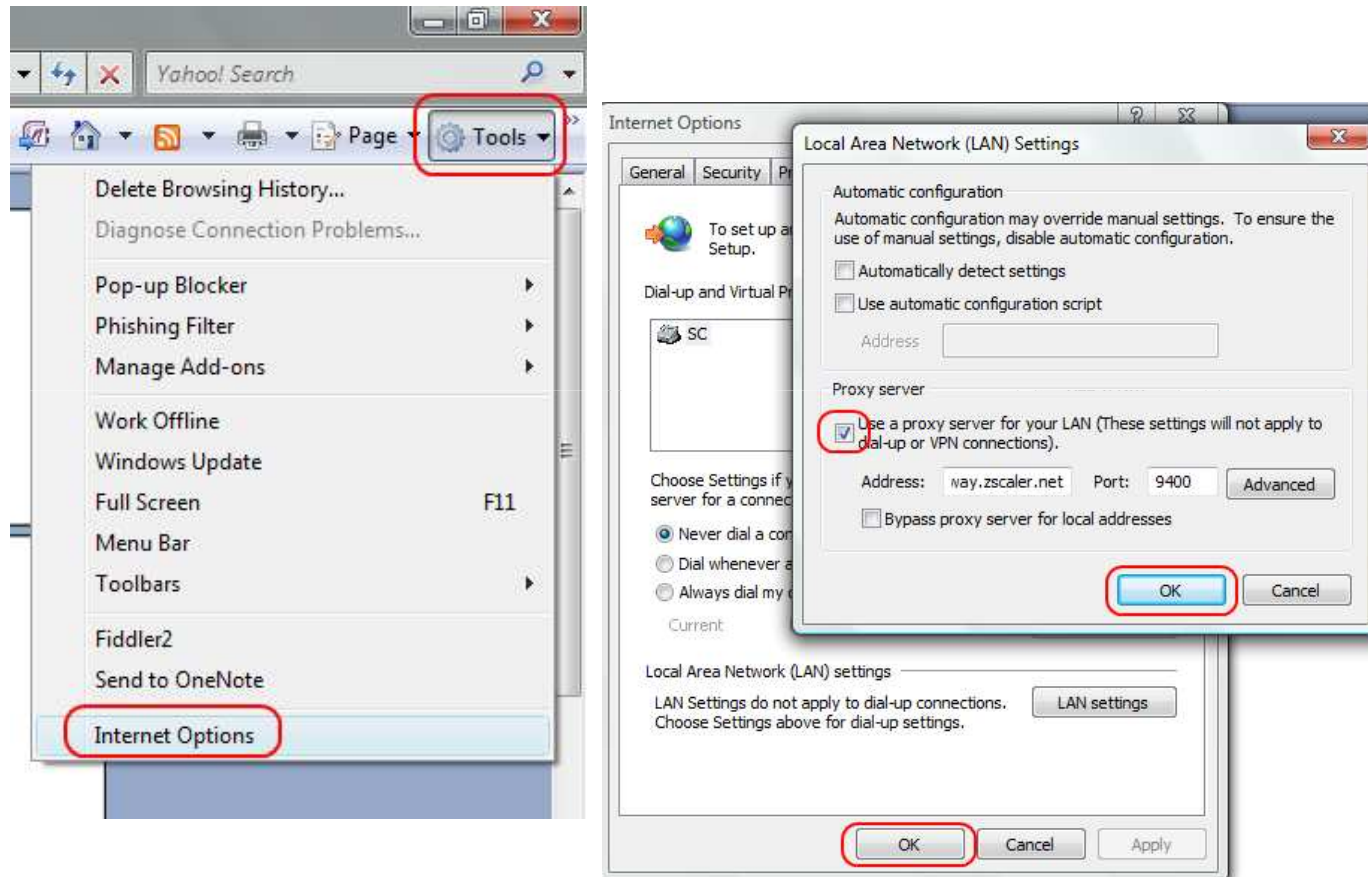
- Users un-doing configuration?
- Alternate browsers?
 - Firefox, Opera, Lunascape* – separate
 - Chrome, Safari – IE
- WPAD/PAC configuration?

Somewhat related commercial

"Mr. Owl, how many ^Clicks does it take to get to
the ~~Tootsie Roll center~~ of a ~~Tootsie Pop~~?"
proxy configuration web browser

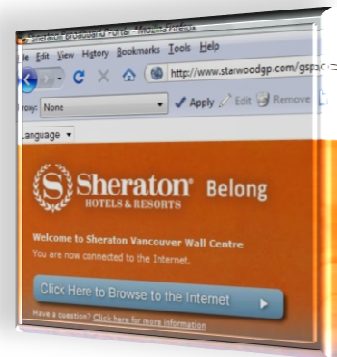


Let's find out...



What about... cont.

- Mobile users?
 - Walled gardens at hotspots/hotels?
 - PAC?
 - WPAD?
- Transparent interception?
 - Outside the org?
 - What ports*?



Let's talk ports

Crowd participation time!

Given hundreds of millions of proxied HTTP transactions over a two month period, how many unique ports would you expect?

Let's talk ports cont.

Our customer traffic from Jan + Feb has HTTP(S) destined to 4400+ different ports, 80+ having significant re-occurring requests

80/443

Top of the list:

1935, 8889, 8080, 8890, 2095, 1375, 9090, 8000, 8001, 9000, 8088,
9393, 81, 82, 8700, 8090, 8081

1935 = Flash streaming media

8889, 8890 = webcams

2095, 9393 = webmail

1375 = live online help SaaS

9000 = AOL webmail + others

81 = alt HTTP, syndication

Let's talk ports cont.

The era of filtering by destination port is gone; meet the new crew of port 443:



YAHOO! VOICE

Moral to the story

For effective traffic inspection and policy enforcement, networks must be (re)designed to eliminate excessive egress opportunities by users.

“Never underestimate the cleverness of users desperate to get to their MySpace/Facebook/Twitter page...”

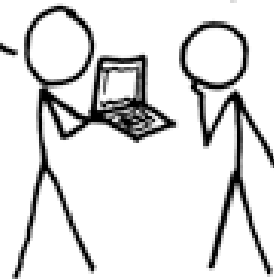
But...

AN IT NERD'S
IMAGINATION:

LET'S RUN P2P OFF THE
CORP TI AND POST ABOUT
IT ON FACEBOOK

NO GO, THE
PROXY FILTERS

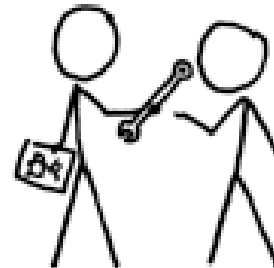
BLAST! OUR
EVIL PLAN
IS FOILED!



WHAT WOULD
ACTUALLY HAPPEN:

CONNECT TO A NEIGHBOR'S
WIRELESS AP TO GET
AROUND THE FILTERING
PROXY.

GOT IT.



Keep in mind...

Once you control your network's egress points, users may turn to:

- Mobile phones (Bluetooth)
- Cellular data dongles ('air cards')
- Open wireless access points of neighboring organizations
- WiMax

If they can get out, stuff can get back in

KThxBye

Jeff.Forrstal@Zscaler.com
Jeff@Forristal.com



Copyright 2009 Zscaler, Inc.