

## Re: [fw-wiz] Blocking Google Talk

---

*Source:* <http://www.derkeiler.com/Mailing-Lists/Firewall-Wizards/2006-06/msg00057.html>

---

- *From:* Devdas Bhagat <[dvb@xxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:dvb@xxxxxxxxxxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Tue, 20 Jun 2006 13:49:07 +0530
- 

On 19/06/06 19:55 -0500, Frank Knobbe wrote:

On Mon, 2006-06-19 at 19:55 -0400, Paul D. Robertson wrote:

It's a reasonable first step. If the user has the ability to modify their resolver configuration, then that may be a bigger issue than running a chat client. [...]

The answer given is enough to enforce the policy from casual abusers, which is really the goal of most protective policy measures. [...]

No, the point is that the answer is a "band-aid" approach that requires a certain setup (the ability to intercept name requests and return fixed IPs). It is not a general solution that anyone can employ, and it requires a more invasive modification of someones network instead of just filtering (or allowing) a port on a firewall.

Bleh. Filtering out nameservers is one way of using a proxy to block traffic. You do run your own recursive resolvers anyway, right?

Not running your own resolvers is a bad idea in the first place.

It is a "band-aid" approach rather than a mature solution. If Google can't provide a mature way of preventing traffic \*1 what does that tell you about the design of the program/protocol?

Lets see, https for authentication, XMPP for communication? Rather open standards, aren't these?

This isn't a bandaid. Oh, and if you really want to stop the problem, why not just prevent the installation of the software in the first place?

Re: [fw-wiz] Blocking Google Talk

Firewalls are bandaids. If software was written correctly, you wouldn't need them in the first place.

With all the stunts modern IM solution perform in order to maintain network connectivity (tunneling even over telnet...sigh), the obvious

And the reason outbound telnet is allowed from a random client host is?

answer is that these protocols are *\*designed\** not to be circumvented or denied. The answer "oh, just modify your network so that name resolution gets forwarded to a central box where you can split requests (like dnscache) and either forward requests to upstream resolvers or provide local responses for the domain in question, and then just return a fake IP address to the client hoping that the OS trusts the DNS servers response enough so that our application gets successfully tricked into not connecting to our servers" ...(/me catching breath after that sentence).... that answer sounds really like a lame duck.

My question would be, why aren't you running your own recursive resolver in the first place? Why are your clients directly talking to the world?

#include <rants/mjr/proxy>

Devdas Bhagat

---

firewall-wizards mailing list  
firewall-wizards@xxxxxxxxxxxxxxxxxxxxxxxxx  
<https://listserv.icsalabs.com/mailman/listinfo/firewall-wizards>