

Tinyproxy

Some of our systems have extra "security needs", and they are not allowed to initiate outgoing connections by default. This means that IP ACLs are used so that they can only reach necessary services (SMTP gateway, DNS resolvers, NTP etc).

Because those hosts do need access to some web sites (mostly for software updates), we use a proxy server to allow them access to those domains.

If you have an IPv6-only host that only needs access to some outside HTTP resources, then this approach kills two birds with one stone:

- Many services are run on CDNs such as Akamai, which renders IP ACLs into a nightmare. A proxy solves this by allowing domains/URLs.
- Some services are only accessible via IPv4 (Microsoft Update, hostupdate.vmware.com, Secunia.com). A dual stack proxy does the protocol translation. If those web sites were the only problems on the IPv6-only system, this is just what you need, and you can avoid using additional complex systems such as NAT64/DNS64.

Because we do not need any caching, but only the access restriction part, I choose tinyproxy because it is very light weight and simple.

It support both IPv4 and IPv6:

```
# This will accept connections on IPv6, but also on IPv4 (IPv4-mapped IPv6 addresses are used)
Listen ::

# This will listen on IPv4 only
Listen 0.0.0.0
# This will listen only on the specified IPv6 address. Not nice, but workable.
Listen 2001:610:148:dead::666
```

Whitelist

I configured tinyproxy to block everything, except a list of domains, by using this configuration:

```
FilterDefaultDeny Yes
Filter "/etc/whitelist"
FilterExtended On
```

Take care when building the white list. While the following entry might look OK and will work OK at first sight:

microsoft.com

It is interpreted as a regular expression, so `roguedomain-not-owned-bymicrosoft.com` will also be accepted.

I wanted a regex to allow:

- `domain.com`
- `subdomain.domain.com`
- `any.number.of.subdomains.domain.com`

Some other sites:

- `s-microsoft.com` as well, as this is used a lot in updates.
- `mstfncsi.com` is a web site used by the Network Connectivity Status Indicator, Windows' network awareness tool (see <http://blog.superuser.com/2011/05/16/windows-7-network-awareness/>).
- Don't forget that systems might access CRLs or OCSP responders, which are hosted on `thawte.com` and `public-trust.com`.

Thus my whitelist look like this:

```

^(.*\.)microsoftupdate\.com$
^(.*\.)msftncsi\.com$
^(ocsp|crt)\.tcs\.terena\.org$
^(.*\.)public-trust\.com$
^crl\.globalsign\.net$
^(.*\.)secunia\.com$
^(.*\.)thawte\.com$
^(.*\.)?(s-)?microsoft\.com$
^(.*\.)usertrust\.com$
^ocsp\.comodoca\.com$
^(.*\.)verisign\.com$
^(.*\.)vmware\.com$
^(.*\.)windowsupdate\.com$
^(api|dellincca|downloads|ftp|www)\.dell\.com$
^www\.adobe\.com$
^update\.exactsoftware\.com$

```

This list is the initial list. By [monitoring the log files](#) you can adjust the list. This is an iterative process, it takes a while to establish a list that is 'right'.

Configuring operating systems and software to use the proxy

Now that you have a proxy, your software should use it as well.

Every OS has its own way of configuring these settings, and not everything is clear from the start. I'll list a few things I ran into while trying to massage everything to use our proxy.

Windows servers

Windows 2003

There are two ways to update software on Windows 2003 (and XP).

1. The "Windows Update" start menu items opens up <http://update.microsoft.com/windowsupdate/v6/default.aspx> in an Internet Explorer browser window. In order for this to work through a proxy, go to Control Panel -> Internet Options. This will bring up the IE settings dialogue go to Connections -> LAN settings, and fill in the stuff there.
2. For automatic updates to work, go to Control Panel -> System -> Automatic Updates, and configure it to your needs (I usually let them install automatically because I don't have the time to look at all the updates, let alone test them. If an update screws up - though luck). The updates downloading is done by [BITS](#), but this does not honour any of the stuff from Internet Options. Proxy settings for BITS are configured using the proxycfg command:

```

C:\Documents and Settings\Administrator>proxycfg -p proxy.terena.org:8888
Microsoft (R) WinHTTP Default Proxy Configuration Tool
Copyright (c) Microsoft Corporation. All rights reserved.
Updated proxy settings
Current WinHTTP proxy settings under:
HKEY_LOCAL_MACHINE\
SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\Connections\
WinHttpSettings :
Proxy Server(s) : proxy.terena.org:8888
Bypass List      : (none)

```

Windows Server 2008R2/2012

There is only one way to configure the updates and that is through the Windows Update control panel.

Similarly to 2003, the proxy settings of BITS need to be set using a command, this time it is done using netsh:

```

C:\Users\Administrator>netsh winhttp set proxy proxy6:8888 "<local>"
Current WinHTTP proxy settings:
Proxy Server(s) : proxy6.terena.org:8888
Bypass List      : (none)

```

There might be other (3rd party) software that uses Internet Explorer to phone home for updates etc. In that case, you need to use the Internet Options control panel again.

In the client version Windows Vista, 7, and 8 this works the same.

Monitoring

To keep an eye on domains that your hosts try to access that are not allowed, run this shell script every morning after the log files have been rotated (7 AM on Ubuntu systems for instance):

```
#!/bin/sh
# Filter PIDs that handled refused domains
TP_LOG="/var/log/tinyproxy/tinyproxy.log"
grep 'Proxying refused' "$TP_LOG" |
sed -r 's/.*(\[[0-9]+\]).*/\1/g' | sort | uniq |
while read pid
do grep "\\$pid" "$TP_LOG"
done |
grep -B2 refused
```

Based on the results, you can add stuff to the white list, or investigate what it going on.